XSA Board V1.1, V1.2 User Manual

How to install, test, and use
your new XSA Board

RELEASE DATE: 12/22/2003

Copyright © 2001-2003 by X Engineering Software Systems Corporation.

All XS-prefix product designations are trademarks of XESS Corp.

All XC-prefix product designations are trademarks of Xilinx.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. Printed in the United States of America.

Table of Contents

1

Preliminaries

Getting Help!

Here are some places to get help if you encounter problems:

- If you can't get the XSA Board hardware to work, send an e-mail message describing your problem to help@xess.com or submit a problem report at http://www.xess.com/help.html. Our web site also has
  - answers to frequently-asked-questions,
  - example designs, application notes and tutorials for the XS Boards,
  - a place to sign-up for our email forum where you can post questions to other XS Board users.
- If you can't get your Xilinx WebPACK software tools installed properly, send an e-mail message describing your problem to hotline@xilinx.com or check their web site at http://www.xilinx.com/support/support.htm.
- If you need help using the WebPACK software to create designs for your XSA Board, then check out this tutorial.

Take notice!!

- The XSA Board requires an external power supply to operate! It does not draw power through the downloading cable from the PC parallel port.
- If you are connecting a 9VDC power supply to your XSA Board, please make sure the center terminal of the plug is positive and the outer sleeve is negative.
- Do not power your XSA Board with a battery! This will not provide enough current to insure reliable operation of the XSA Board.

Packing List

Here is what you should have received in your package:

- an XSA Board;
- a 6' cable with a 25-pin male connector on each end;
- an XSTOOLS CDROM with software utilities and documentation for using the XSA Board.

2

Installation

Installing the XSTOOLS Utilities and Documentation

Xilinx currently provides the WebPACK tools for programming their CPLDs and Spartan-II FPGAs. The XESS CDROM contains a version of WebPACK that will generate bitstream configuration files compatible with your XSA Board. You can also download the most current version of the WebPACK tools from the Xilinx website..

In addition, XESS Corp. provides the XSTOOLS utilities for interfacing a PC to your XSA Board. Run the SETUP.EXE program on the XSTOOLS CDROM to install these utilities.

Applying Power to Your XSA Board

You can use your XSA Board in three ways, distinguished by the method you use to apply power to the board.

Using a 9VDC wall-mount power supply

You can use your XSA Board all by itself to experiment with logic designs. Just place the XSA Board on a non-conducting surface as shown in Figure 1. Then apply power to jack J5 of the XSA Board from a 9V DC wall-mount power supply with a 2.1 mm female, center-positive plug. (See Figure 2 for the location of jack J5 on your XSA Board.) The on-board voltage regulation circuitry will create the voltages required by the rest of the XSA Board circuitry. **Be careful!! The voltage regulators on the XSA Board will become hot.** Attach a heat sink to them if necessary.

Powering Through the PS/2 Connector

You can use your XSA Board with a laptop PC by connecting a PS/2 male-to-male cable from the PS/2 port of the laptop to the J4 connector. You must also have a shunt across pins 1 and 2 of jumper J7. The on-board voltage regulation circuitry will create the voltages required by the rest of the XSA Board circuitry. **Many PS/2 ports cannot supply more than 0.5A so large, fast FPGA designs may not work when using this power source!**

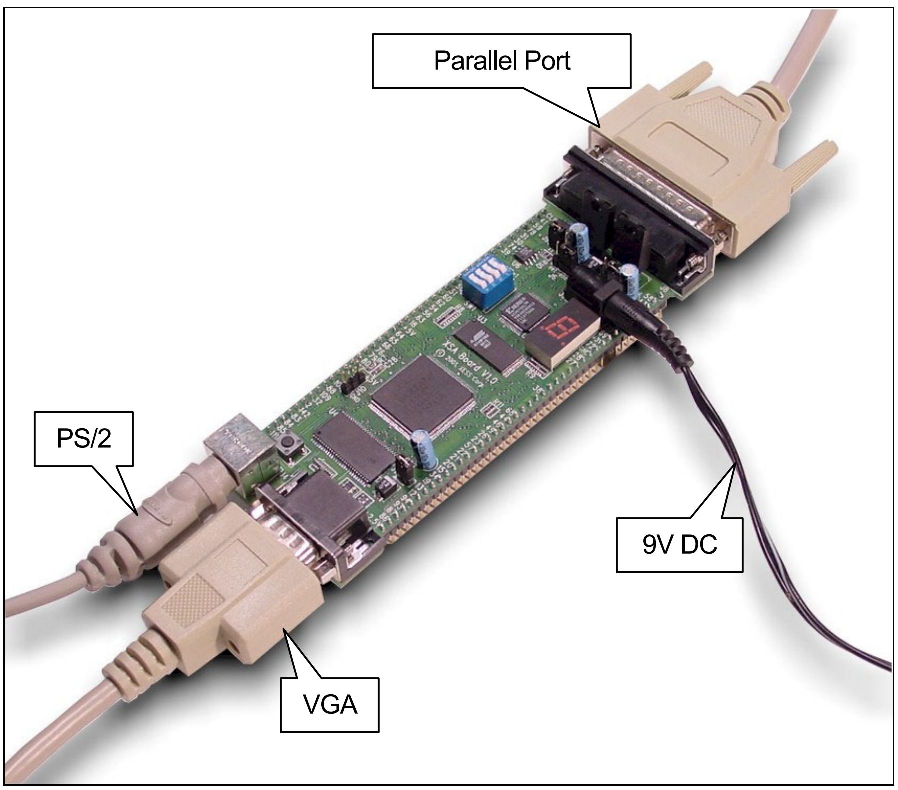
Solderless Protoboard Installation

The two rows of pins from your XSA Board can be plugged into a solderless protoboard with holes spaced at 0.1" intervals. (One of the A.C.E. protoboards from 3M is a good choice.) Once plugged in, many of the pins of the FPGA are accessible to other circuits on the protoboard. (The numbers printed next to the rows of pins on your XSA Board

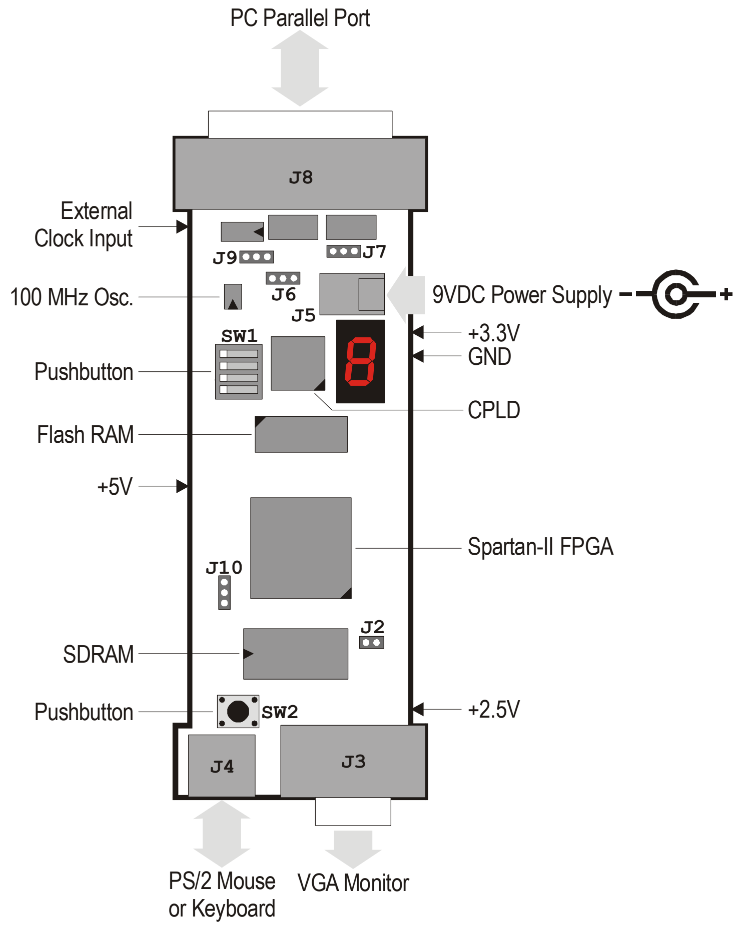
correspond to the pin numbers of the FPGA.) Power can still be supplied to your XSA Board though jack J5, or power can be applied directly through several pins on the underside of the board. Just connect +5V, +3.3V, +2.5V and ground to the pins of your XSA Board listed in Table 1.

• Table 1: Power supply pins for the XSA Board.

Voltage	Pin	Note
+5V	2	
+3.3V	22	Remove the shunt from jumper J7 if you wish to use your own +3.3V supply. Leave the shunt on jumper J7 to generate the +3.3V supply from the +5V supply.
+2.5V	54	Remove the shunt from jumper J2 if you wish to use your own +2.5V supply. Leave the shunt on jumper J2 to generate the +2.5V supply from the +3.3V supply.
GND	52	



• Figure 1: External connections to the XSA Board.



• Figure 2: Arrangement of components on the XSA Board.

Connecting a PC to Your XSA Board

The 6' DB25 male-to-male cable included with your XSA Board connects it to a PC. One end of the cable attaches to the parallel port on the PC and the other connects to the female DB-25 connector (J8) at the top of the XSA Board as shown in Figure 1.

Connecting a VGA Monitor to Your XSA Board

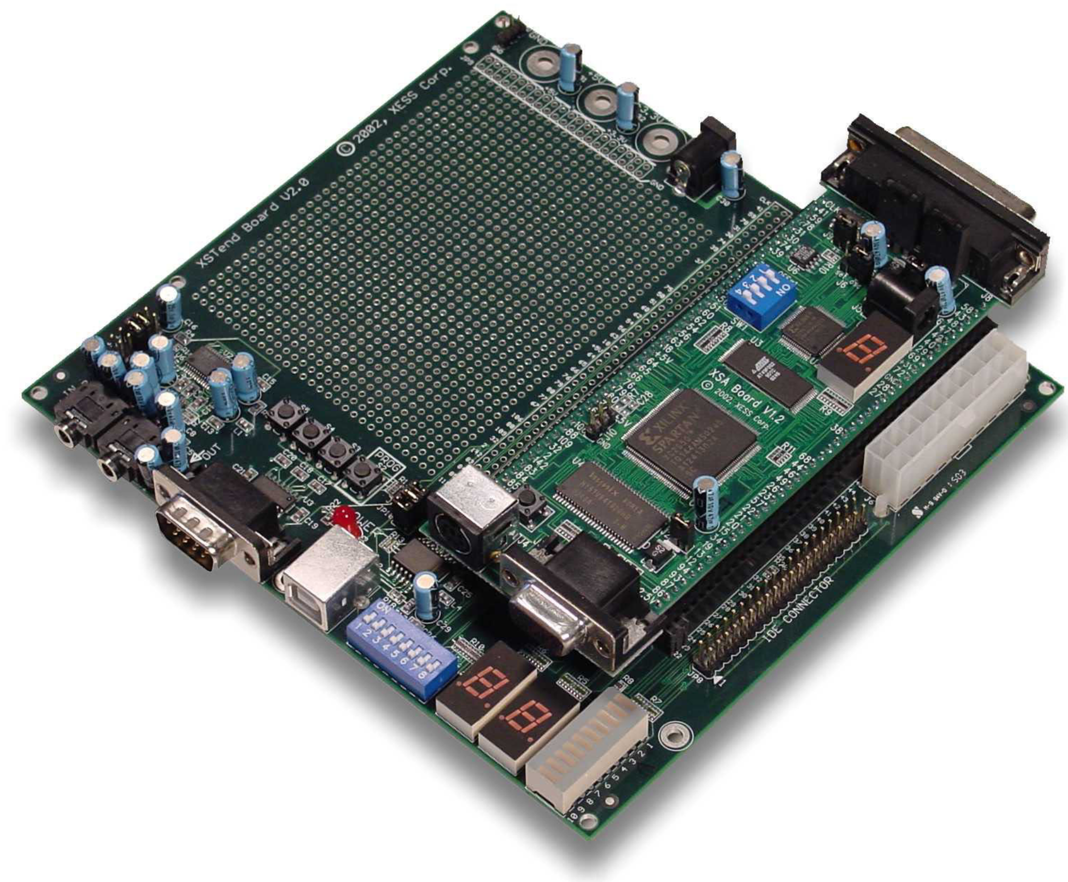
You can display images on a VGA monitor by connecting it to the 15-pin J3 connector at the bottom of your XSA Board (see Figure 1). You will have to create a VGA driver circuit for your XSA Board to actually display an image. You can find an example VGA driver at http://www.xess.com/ho03000.html.

Connecting a Mouse or Keyboard to Your XSA Board

You can accept inputs from a keyboard or mouse by connecting it to the J4 PS/2 connector at the bottom of your XSA Board (see Figure 1). You can find an example keyboard driver at http://www.xess.com/ho03000.html.

Inserting the XSA Board into an XStend Board

If you purchased the optional XST-2.*x* Board, then the XSA Board is inserted as shown below. Refer to the XST-2.*x* Board Manual for more details.



Setting the Jumpers on Your XSA Board

The default jumper settings shown in Table 2 configure your XSA Board for use in a logic design environment. You will need to change the jumper settings only if you are:

- downloading FPGA bitstreams to your XSA Board using the Xilinx iMPACT software;

- reprogramming the clock frequency on your XSA Board (see page 11);
- changing the power sources for the XSA supply voltages.

• Table 2: Jumper settings for XSA Boards.

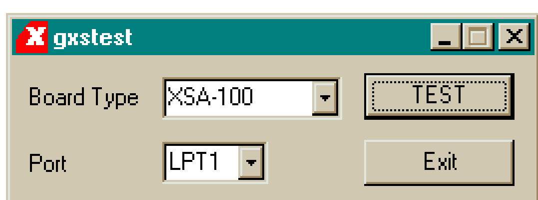
Jumper	Setting	Purpose
J2	On (default)	A shunt should be installed if the +2.5V supply voltage is derived from the +3.3V supply.
	Off	The shunt should be removed if the +2.5V supply voltage is applied from an external source through pin 22 of the XSA Board (labeled "+2.5V" at the lower right-hand corner of the board).
J6	1-2 (set)	The shunt should be installed on pins 1 and 2 (set) when setting the frequency of the programmable oscillator.
	2-3 (osc) (default)	The shunt should be installed on pins 2 and 3 (osc) during normal operations when the programmable oscillator is generating a clock signal.
J7	1-2 (default)	The shunt should be installed on pins 1 and 2 if the +3.3V supply voltage is derived from the +5V supply.
	2-3	The shunt should be installed on pins 2 and 3 if the +3.3V supply voltage is derived from the 9VDC supply applied through jack J5.
J9	1-2 (xi)	The shunt should be installed on pins 1 and 2 (xi) if the XSA Board is to be downloaded using the Xilinx iMPACT software.
	2-3 (xs) (default)	The shunt should be installed on pins 2 and 3 (xs) if the XSA Board is to be downloaded using the XESS GXSLOAD software.
J10	N/A	This is a header that provides access to the +5V and GND references on the board. No shunt should be placed on this header.

Testing Your XSA Board

Once your XSA Board is installed and the jumpers are in their default configuration, you can test the board using the GUI-based GXSTEST utility as follows.

You start GXSTEST by clicking on the 

icon placed on the desktop during the XSTOOLS installation. This brings up the window shown below.



Next you select the parallel port that your XSA Board is connected to from the Port pulldown list. GXSTEST starts with parallel port LPT1 as the default, but you can also select LPT2 or LPT3 depending upon the configuration of your PC.

After selecting the parallel port, you select either the XSA-50 or XSA-100 item in the Board Type pulldown list. Then click on the TEST button to start the testing procedure. GXSTEST will configure the FPGA to perform a test procedure on your XSA Board.

Within thirty seconds you will see a **O** displayed on the LED digit if the test completes successfully. Otherwise an **E** will be displayed if the test fails. A status window will also appear on your PC screen informing you of the success or failure of the test.

If your XSA Board fails the test, you will be shown a checklist of common causes for failure. If none of these causes applies to your situation, then test the XSA Board using another PC. In our experience, 99.9% of all problems are due to the parallel port. If you cannot get your board to pass the test even after taking these steps, then contact XESS Corp for further assistance.

As a result of testing the XSA Board, the CPLD is programmed with the standard parallel port interface found in the dwnldpar.svf bitstream file located within the XSTOOLS\XSA folder. This is the standard interface that should be loaded into the CPLD when you want to use it with the GXSLOAD utility.

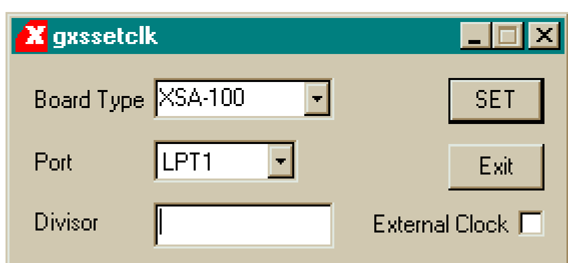
Setting the XSA Board Clock Oscillator Frequency

The XSA Board has a 100 MHz programmable oscillator (a Dallas Semiconductor DS1075Z-100). The 100 MHz master frequency can be divided by factors of 1, 2, ... up to 2052 to get clock frequencies of 100 MHz, 50 MHz, ... down to 48.7 KHz, respectively. The divided frequency is sent to the rest of the XSA Board circuitry as a clock signal.

The divisor is stored in non-volatile memory in the oscillator chip so it will resume operation at its programmed frequency whenever power is applied to the XSA Board. You can store a particular divisor into the oscillator chip by using the GUI-based GXSSETCLK as follows.

You start GXSSETCLK by clicking on the 

icon placed on the desktop during the XSTOOLS installation. This brings up the window shown below.



Your next step is to select the parallel port that your XSA Board is connected to from the Port pulldown list. Then select either XSA-50 or XSA-100 in the Board Type pulldown list.

Next you enter a divisor between 1 and 2052 into the Divisor text box and then click on the SET button. Then follow the sequence of instructions given by XSSETCLK for moving shunts and removing and restoring power during the oscillator programming process. At the completion of the process, the new frequency will be programmed into the DS1075.

An external clock signal can be substituted for the internal 100 MHz oscillator of the DS1075. Checking the External Clock checkbox will enable this feature in the

programmable oscillator chip. If this option is selected, you are then responsible for providing the external clock to the XSA Board through pin 64 (labeled “CLK” at the upper left-hand corner of the board).

3

Programming

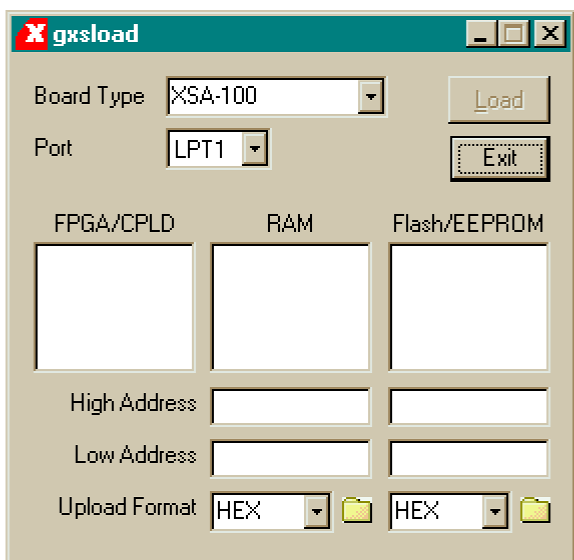
This section will show you how to download a logic designs into the FPGA and CPLD of your XSA Board and how to download and upload data to and from the SDRAM and Flash devices on the board.

Downloading Designs into the FPGA and CPLD of Your XSA Board

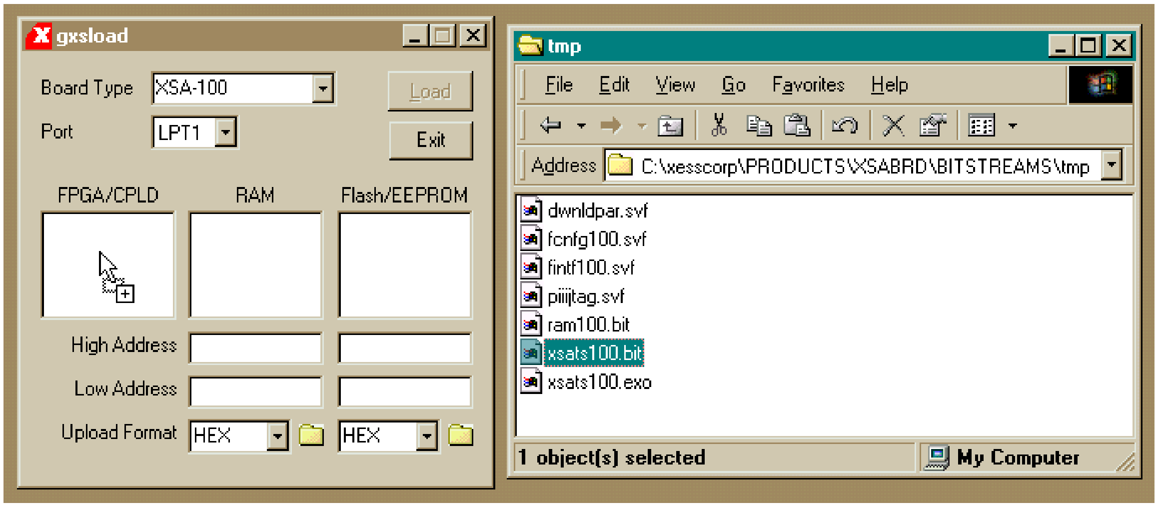
During the development and testing phases, you will usually connect the XSA Board to the parallel port of a PC and download your circuit each time you make changes to it. You can download a Spartan-II FPGA design into your XSA Board using the GXSLOAD utility as follows.

You start GXSLOAD by clicking on the 

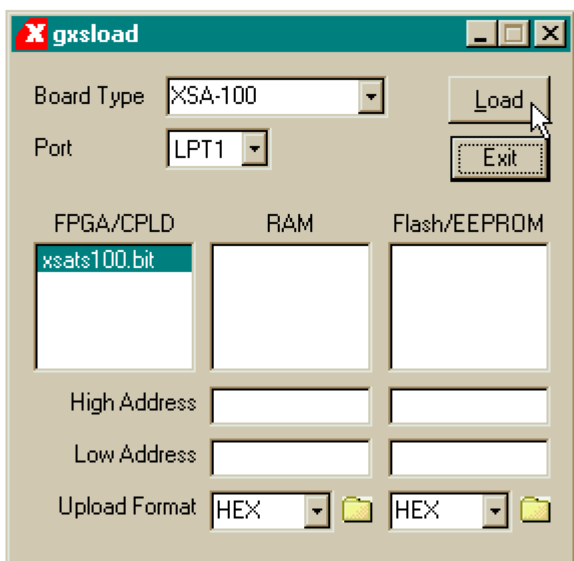
icon placed on the desktop during the XSTOOLS installation. This brings up the window shown below. Then select the type of XS Board you are using and the parallel port to which it is connected as follows.



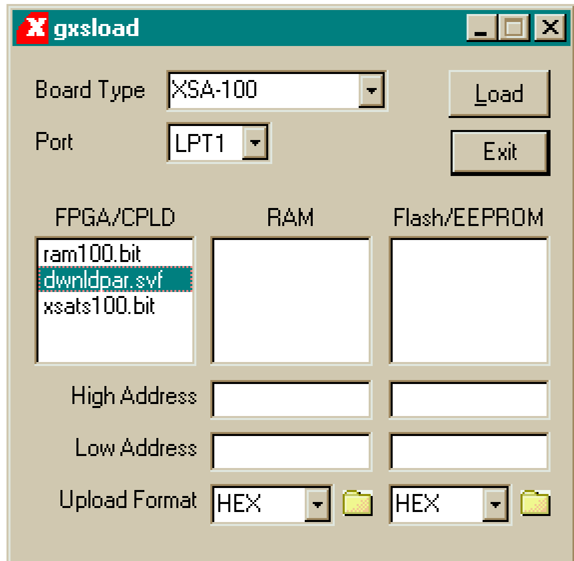
After setting the board type and parallel port, you can download .BIT or .SVF files to the Spartan-II FPGA or XC9572XL CPLD on your XSA Board simply by dragging them to the FPGA/CPLD area of the GXSLOAD window as shown below.



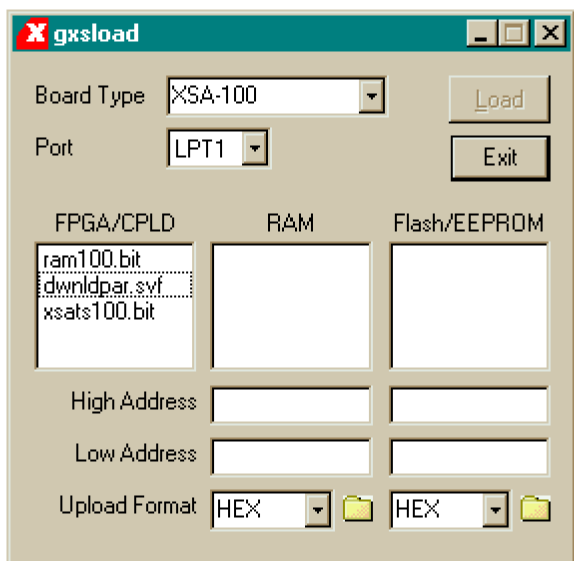
Once you release the left mouse button and drop the file, the highlighted file name appears in the FPGA/CPLD area and the Load button in the GXSLOAD window is enabled. Clicking on the Load button will begin sending the highlighted file to the XSA Board through the parallel port connection. .BIT files contain configuration bitstreams that are loaded into the FPGA while .SVF files will go to the CPLD. GXSLOAD will reject any non-downloadable files (ones with a suffix other than .BIT or .SVF). During the downloading process, GXSLOAD will display the name of the file and the progress of the current download.



You can drag & drop multiple files into the FPGA/CPLD area. Clicking your mouse on a filename will highlight the name and select it for downloading. Only one file at a time can be selected for downloading.



Double-clicking the highlighted file will deselect it so no file will be downloaded Doing this disables the Load button.



Storing Non-Volatile Designs in Your XSA Board

The Spartan-II FPGA on the XSA Board stores its configuration in an on-chip SRAM which is erased whenever power is removed. Once your design is finished, you may want to store the bitstream in the 256-KByte Flash device on the XSA Board which configures the FPGA for operation as soon as power is applied.

Before downloading to the Flash, the FPGA .BIT file must be converted into a .EXO or .MCS format using one of the following commands:

```
promgen –u 0 file.bit –p exo –s 256

promgen –u 0 file.bit –p mcs –s 256
```

In the commands shown above, the bitstream in the file.bit file is transformed into an .EXO or .MCS file format starting at address zero and proceeding upward until an upper limit of 256 KBytes is reached.

Before attempting to program the Flash, you must place all four DIP switches into the OFF position!

After the .EXO or .MCS file is generated, it is loaded into the Flash device by dragging it into the Flash/EEPROM area and clicking on the Load button. This activates the following sequence of steps:

1. The entire Flash device is erased.

2. The CPLD on the XSA Board is reprogrammed to create an interface between the Flash device and the PC parallel port. (This interface is stored in the fintf100.svf bitstream file located within the XSTOOLS\XSA folder.)

3. The contents of the .EXO or .MCS file are downloaded into the Flash through the parallel port.

4. The CPLD is reprogrammed to create a circuit that configures the FPGA with the contents of the Flash when power is applied to the XSA Board. (This configuration loader is stored in the fcnfg.svf bitstream file located within the XSTOOLS\XSA folder.)

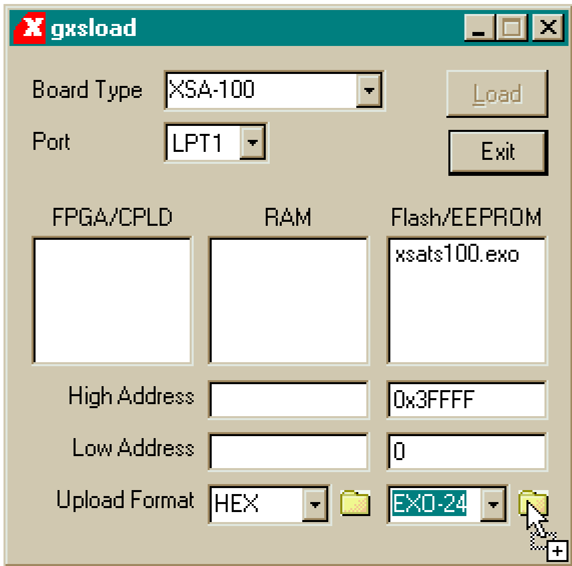
Multiple files can be stored in the Flash device just by dragging them into the Flash/EEPROM area, highlighting the files to be downloaded and clicking the Load button. (Note that anything previously stored in the Flash will be erased by each new download.) This is useful if you need to store information in the Flash in addition to the FPGA bitstream. Files are selected and de-selected for downloading just by clicking on their names in the Flash/EEPROM area. **The address ranges of the data in each file should not overlap or this will corrupt the data stored in the Flash device!**

You can also examine the contents of the Flash device by uploading it to the PC. To upload data from an address range in the Flash, type the upper and lower bounds of the range into the High Address and Low Address fields below the Flash/EEPROM area, and select the format in which you would like to store the data using the Upload Format pulldown list. Then click on the file icon and drag & drop it into any folder. This activates the following sequence of steps:

1. The CPLD on the XSA Board is reprogrammed to create an interface between the Flash device and the PC parallel port.

2. The Flash data between the high and low addresses (inclusive) is uploaded through the parallel port.

3. The uploaded data is stored in a file named FLSHUPLD with an extension that reflects the file format.



The uploaded data can be stored in the following formats:

MCS: Intel hexadecimal file format. This is the same format generated by the promgen utility with the –p mcs option.

HEX: Identical to MCS format.

EXO-16: Motorola S-record format with 16-bit addresses (suitable for 64 KByte uploads only).

EXO-24: Motorola S-record format with 24-bit addresses. This is the same format generated by the promgen utility with the –p exo option.

EXO-32: Motorola S-record format with 32-bit addresses.

XESS-16: XESS hexadecimal format with 16-bit addresses. (This is a simplified file format that does not use checksums.)

XESS-24: XESS hexadecimal format with 24-bit addresses.

XESS-32: XESS hexadecimal format with 32-bit addresses.

After the data is uploaded from the Flash, the CPLD on the XSA Board is left with the Flash interface programmed into it. You will need to reprogram the CPLD with either the parallel port or Flash configuration circuit before the board will function again. The CPLD configuration bitstreams are stored in the following files:

XSTOOLS\XSA\dwnldpar.svf: Drag & drop this file into the FPGA/CPLD area and click on the Load button to put the XSA in a mode where it will configure the FPGA through the parallel port.

XSTOOLS\XSA\ fcnfg.svf: Drag & drop this file into the FPGA/CPLD area and click on the Load button to put the XSA in a mode where it will configure the FPGA with the contents of the Flash device upon power-up.

Downloading and Uploading Data to the SDRAM in Your XSA Board

The XSA-100 Board contains a 16-MByte synchronous DRAM (8M x 16 SDRAM) whose contents can be downloaded and uploaded by GXSLOAD. (The XSA-50 has an 8-MByte SDRAM organized as 4M x 16.) This is useful for initializing the SDRAM with data for use by the FPGA and then reading the SDRAM contents after the FPGA has operated upon it. The SDRAM is loaded with data by dragging & dropping one or more .EXO, .MCS, .HEX, and/or .XES files into the RAM area of the GXSLOAD window and then clicking on the Load button. This activates the following sequence of steps:

1. The Spartan-II FPGA on the XSA Board is reprogrammed to create an interface between the RAM device and the PC parallel port. (This interface is stored in the ram100.bit or ram50.bit bitstream file located within the XSTOOLS\XSA folder. **The CPLD must have previously been loaded with the dwnldpar.svf file found in the same folder.**)

2. The contents of the .EXO, .MCS, .HEX or .XES files are downloaded into the SDRAM through the parallel port. **The data in the files will overwrite each other if their address ranges overlap.**

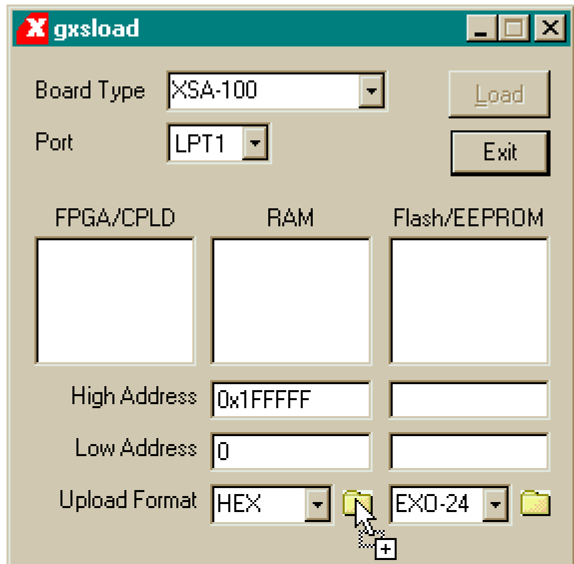
3. If any file is highlighted in the FPGA/CPLD area, then this bitstream is loaded into the FPGA or CPLD on the XSA Board. Otherwise the FPGA remains configured as an interface between the PC and the SDRAM.

You can also examine the contents of the SDRAM device by uploading it to the PC. To upload data from an address range in the SDRAM, type the upper and lower bounds of the range into the High Address and Low Address fields below the RAM area, and select the format in which you would like to store the data using the Upload Format pulldown list. Then click on the file icon and drag & drop it into any folder. This activates the following sequence of steps:

1. The Spartan-II FPGA on the XSA Board is reprogrammed to create an interface between the RAM device and the PC parallel port. (This interface is stored in the ram100.bit or ram50.bit bitstream file located within the XSTOOLS\XSA folder.)

2. The SDRAM data between the high and low addresses (inclusive) is uploaded through the parallel port.

3. The uploaded data is stored in a file named RAMUPLD with an extension that reflects the file format.



The 16-bit data words in the SDRAM are mapped into the eight-bit data format of the .HEX, .MCS, .EXO and .XES files using a Big Endian style. That is, the 16-bit word at address *N* in the SDRAM is stored in the eight-bit file with the upper eight bits at location *2N* and the lower eight bits at location *2N+1*. This byte-ordering applies for both RAM uploads and downloads.

4

Programmer's Models

This section describes the various sections of the XSA Board and shows how the I/O of the FPGA and CPLD are connected to the rest of the circuitry. The schematics which follow are less detailed so as to simplify the descriptions. Please refer to the complete schematics at the end of this document if you need more details.

XSA Board Organization

The XSA Board contains the following components:

XC2S50 or XC2S100 Spartan-II FPGA: This is the main repository of programmable logic on the XSA Board.

XC9572XL CPLD: This CPLD manages the interface between the PC parallel port and the rest of the XSA Board.

Osc: A programmable oscillator generates the master clock for the XSA Board.

Flash: A 128 or 256-KByte Flash device provides non-volatile storage for data and configuration bitstreams.

SDRAM: An 8 or 16-MByte SDRAM provides volatile data storage accessible by the FPGA.

LED: A seven-segment LED allows visible feedback as the XSA Board operates.

DIP switch: A four-position DIP switch passes settings to the XSA Board or controls the upper address bits of the Flash device.

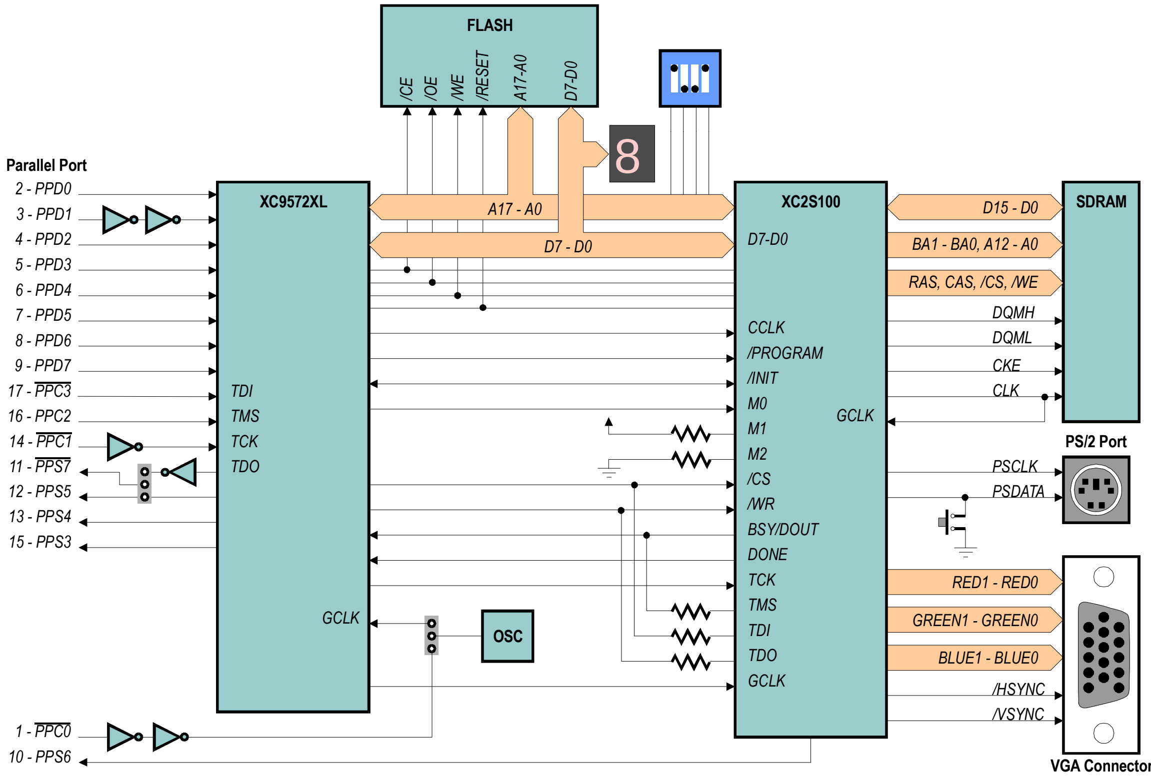
Pushbutton: A single pushbutton sends momentary contact information to the FPGA.

Parallel Port: This is the main interface for passing configuration bitstreams and data to and from the XSA Board.

PS/2 Port: A keyboard or mouse can interface to the XSA Board through this port.

VGA Port: The XSA Board can send signals to display graphics on a VGA monitor through this port.

Prototyping Header: Many of the FPGA I/O pins are connected to the 84 pins on the bottom of the XSA Board that are meant to mate with solderless breadboards.



• Figure 3: XSA Board programmer's model.

Programmable logic: Spartan-II FPGA and XC9572XL CPLD

The XSA Board contains two programmable logic chips:

- A 50-Kgate XC2S50 or 100-Kgate Xilinx XC2S100 Spartan-II FPGA in a 144-pin PQFP package. The FPGA is the main repository of programmable logic on the XSA Board.

- A Xilinx XC9572XL CPLD is used to manage the configuration of the FPGA via the parallel port. The CPLD also controls the programming of the Flash RAM on the XSA Board.

100 MHz Programmable Oscillator

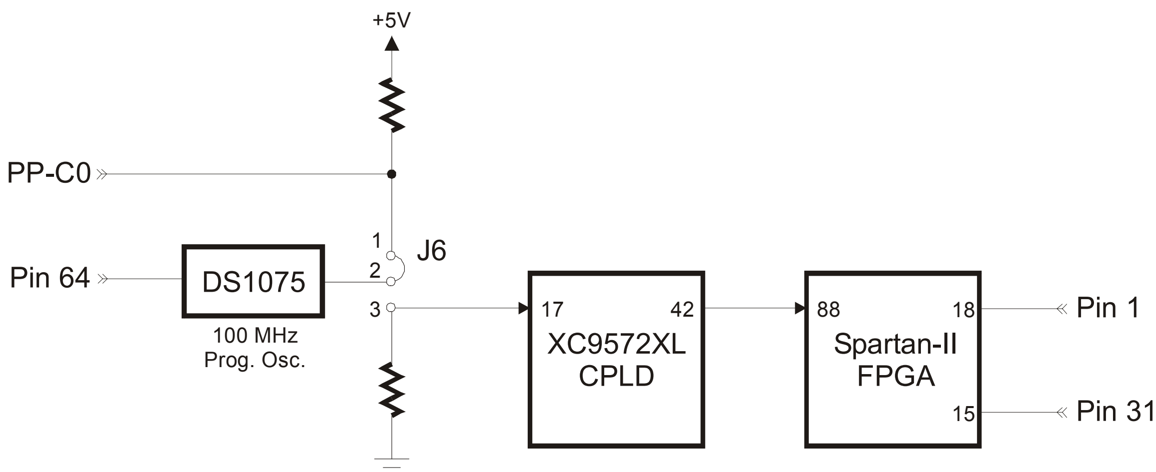
A Dallas DS1075 programmable oscillator provides a clock signal to both the FPGA and the CPLD. The DS1075 has a maximum frequency of 100 MHz that is divided to provide frequencies of 100 MHz, 50 MHz, 33.3 MHz, 25 MHz, ..., 48.7 KHz. The clock signal from the DS1075 is connected to a dedicated clock input of the CPLD. The CPLD passes the clock signal on to the FPGA. This allows the CPLD to control the clock source for the FPGA.

To set the divisor value, the DS1075 must be placed in its programming mode. This is done by pulling the clock output to +5V on power-up with a shunt across pins 1 and 2 of jumper J6. Then programming commands to set the divisor are sent to the DS1075 through control pin C0 of the parallel port. The divisor is stored in EEPROM in the DS1075 so it will be retained even when power is removed from the XSA Board.

The shunt on jumper J6 must be across pins 2 and 3 to make the oscillator output a clock signal upon power-up. The clock signal enters a dedicated clock input of the CPLD. Then the CPLD can output a clock signal to a dedicated clock input of the FPGA.

To get a precise frequency value or to sync the XSA circuitry with an external system, you can insert an external clock signal of up to 50 MHz through pin 64 of the prototyping header. This external clock takes the place of the internal 100 MHz clock source in the DS1075 oscillator. You must use the GXSSETCLK software utility to enable the external clock input of the DS1075.

Clock signals can also be directly applied to two of the dedicated clock inputs of the FPGA through the pins of the prototyping header.

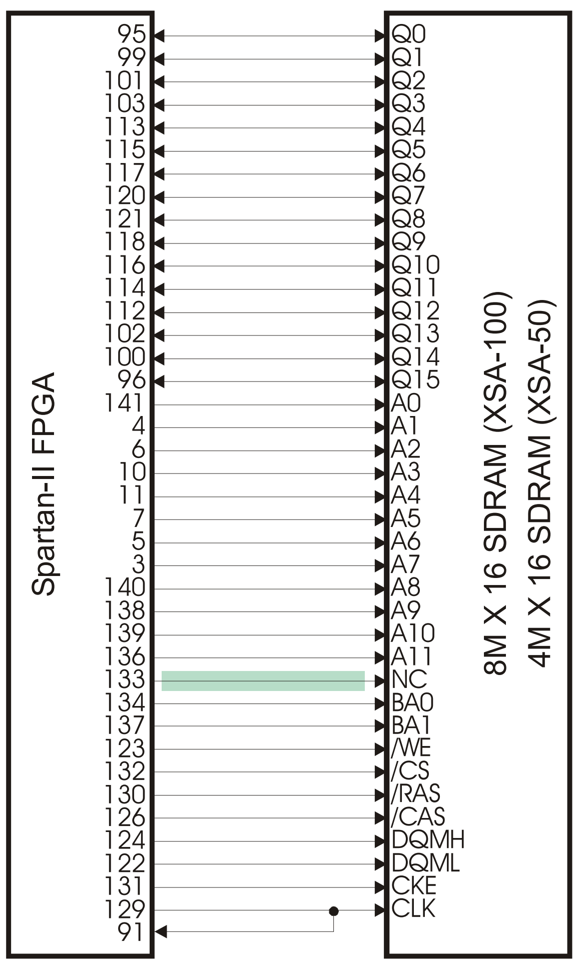


Synchronous DRAM

The various SDRAM organizations and manufacturers used on the XSA Boards are given in the following table.

Board	SDRAM	
	Organization	Manufacturer & Part No.
XSA-50	4M x 16	Hynix HY57V641620HGT-H
	4M x 16	Samsung K4S641632F-TC75000
XSA-100	8M x 16	Hynix HY57V281620HCT-H
	8M x 16	Samsung K4S281632E-TC75000

The SDRAM is connected to the FPGA as shown below. Currently, FPGA pin 133 drives a no-connect pin of the SDRAM but this could be used in the future as the thirteenth row/column address bit of a larger SDRAM. Also, the SDRAM clock signal is re-routed back to a dedicated clock input of the FPGA to allow synchronization of the FPGA's internal operations with the SDRAM operations.

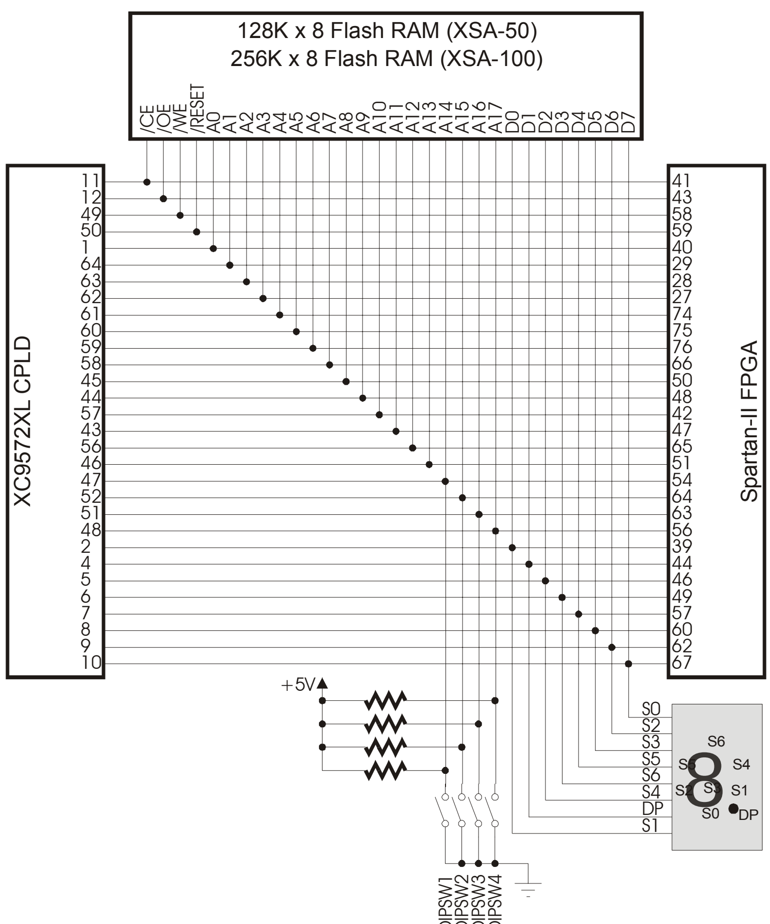


Flash RAM

The Flash RAM organizations and manufacturer used on the XSA Boards are given in the following table.

Board	Flash RAM	
	Organization	Manufacturer & Part No.
XSA-50	128K x 8	Atmel AT49F001 Flash RAM
XSA-100	256K x 8	Atmel AT49F002 Flash RAM

The Flash RAM is connected so both the FPGA and CPLD have access. Typically, the CPLD will program the Flash with data passed through the parallel port. If the data is an FPGA configuration bitstream, then the CPLD can be configured to program the FPGA with the bitstream from Flash whenever the XSA Board is powered up. (See the application note XSA Flash Programming and SpartanII Configuration for more details on this.) After power-up, the FPGA can read and/or write the Flash. (Of course, the CPLD and FPGA have to be programmed such that they do not conflict if both are trying to access the Flash.) The Flash is disabled by raising the /CE pin to a logic 1 thus making the I/O lines connected to the Flash available for general-purpose communication between the FPGA and the CPLD.



Seven-Segment LED

The XSA Board has a 7-segment LED digit for use by the FPGA or the CPLD. The segments of this LED are active-high meaning that a segment will glow when a logic-high is applied to it.

The LED shares the same pins as the eight bits of the Flash RAM data bus.

Four-Position DIP Switch

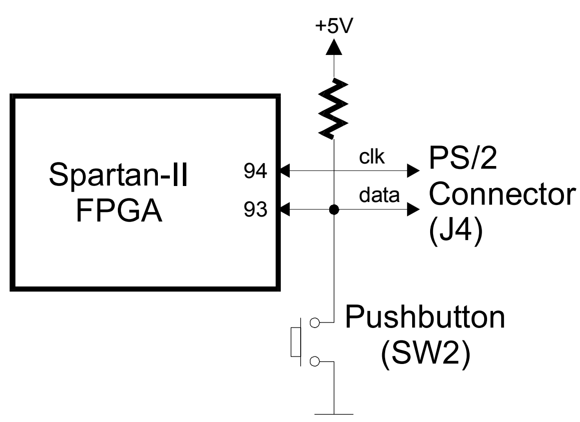
The XSA Board has a bank of four DIP switches accessible from the CPLD and FPGA. When closed or ON, each switch pulls the connected pin of the FPGA and CPLD to ground. Otherwise, the pin is pulled high through a resistor when the switch is open or OFF.

When not being used, the DIP switches should be left in the open or OFF configuration so the pins of the FPGA and CPLD are not tied to ground and can freely move between logic low and high levels.

The DIP switches also share the same pins as the uppermost four bits of the Flash RAM address bus. If the Flash RAM is programmed with several FPGA bitstreams, then the DIP switches can be used to select a particular bitstreams which will be loaded into the FPGA by the CPLD on power-up. However, this feature is not currently supported by the CPLD configuration that loads the FPGA from the Flash RAM (XSTOOLS\XSA\fcnfg.svf).

PS/2 Port

The XSA Board provides a PS/2-style interface (mini-DIN connector J4) to either a keyboard or a mouse. The FPGA receives two signals from the PS/2 interface: a clock signal and a serial data stream that is synchronized with the falling edge of the clock.

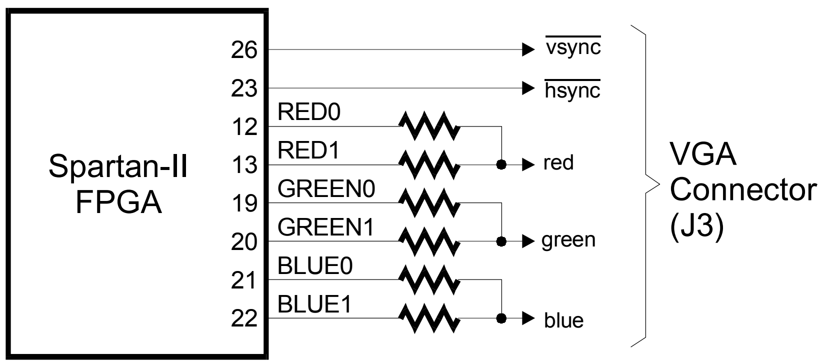


Pushbutton

The XSA Board has a single pushbutton that shares the FPGA pin connected to the data line of the PS/2 port. The pushbutton applies a low level to the FPGA pin when pressed and a resistor pulls the pin to a high level when the pushbutton is not pressed.

VGA Monitor Interface

The FPGA can generate a video signal for display on a VGA monitor. When the FPGA is generating VGA signals, the FPGA outputs two bits each of red, green, and blue color information to a simple resistor-ladder DAC. The outputs of the DAC are sent to the RGB inputs of a VGA monitor along with the horizontal and vertical sync pulses (/HSYNC, /VSYNC) from the FPGA.



Parallel Port Interface

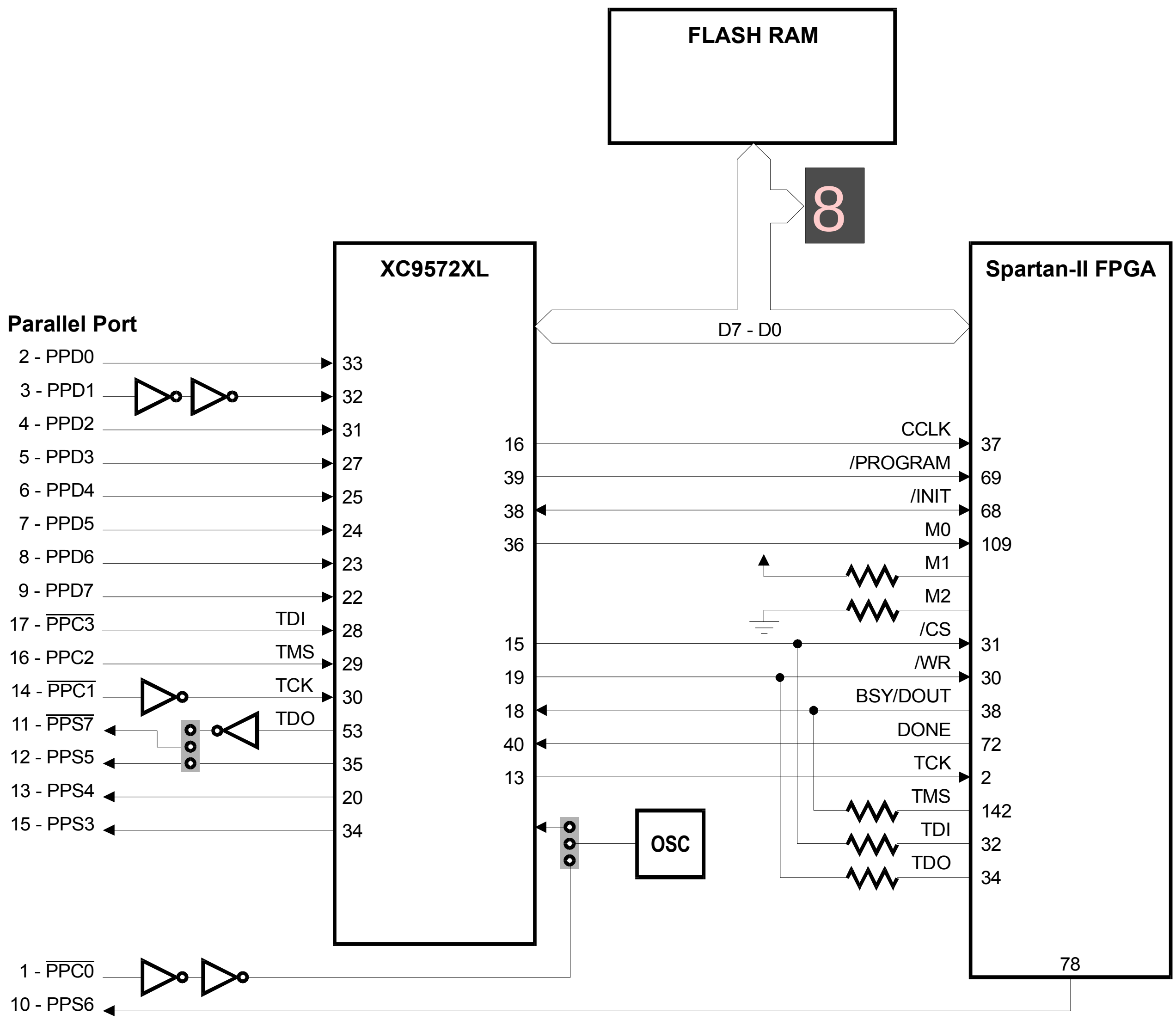
The parallel port is the main interface for communicating with the XSA Board. Control line C0 goes directly to the DS1075 oscillator and is used for setting the divisor as described previously, and status line S6 connects directly to the FPGA for use as a communication line from the FPGA back to the PC. The CPLD handles the fifteen remaining active lines of the parallel port as follows.

Three of the parallel port control lines, C1–C3, connect to the JTAG pins through which the CPLD is programmed. The C1 control line clocks configuration data presented on the C3 line into the CPLD while the C2 signal steers the actions of the CPLD programming state machine. Meanwhile, information from the CPLD returns to the PC through status line S7.

The eight data lines, D0–D7, and the remaining three status lines, S3–S5, connect to general-purpose pins of the CPLD. The CPLD can be programmed to act as an interface between the FPGA and the parallel port (the dwnldpar.svf file is an example of such an interface). Schmitt-trigger inverters are inserted into the D1 line so it can carry a clean clock edge for use by any state machine programmed into the CPLD. The CPLD connects to the configuration pins of the Spartan-II FPGA so it can pass bitstreams from the parallel port to the FPGA. The actual configuration data is presented to the FPGA on the same 8-bit bus that also connects to the Flash RAM and seven-segment LED. The CPLD also drives the configuration pins (CCLK, /PROGRAM, /CS, and /WR) of the FPGA

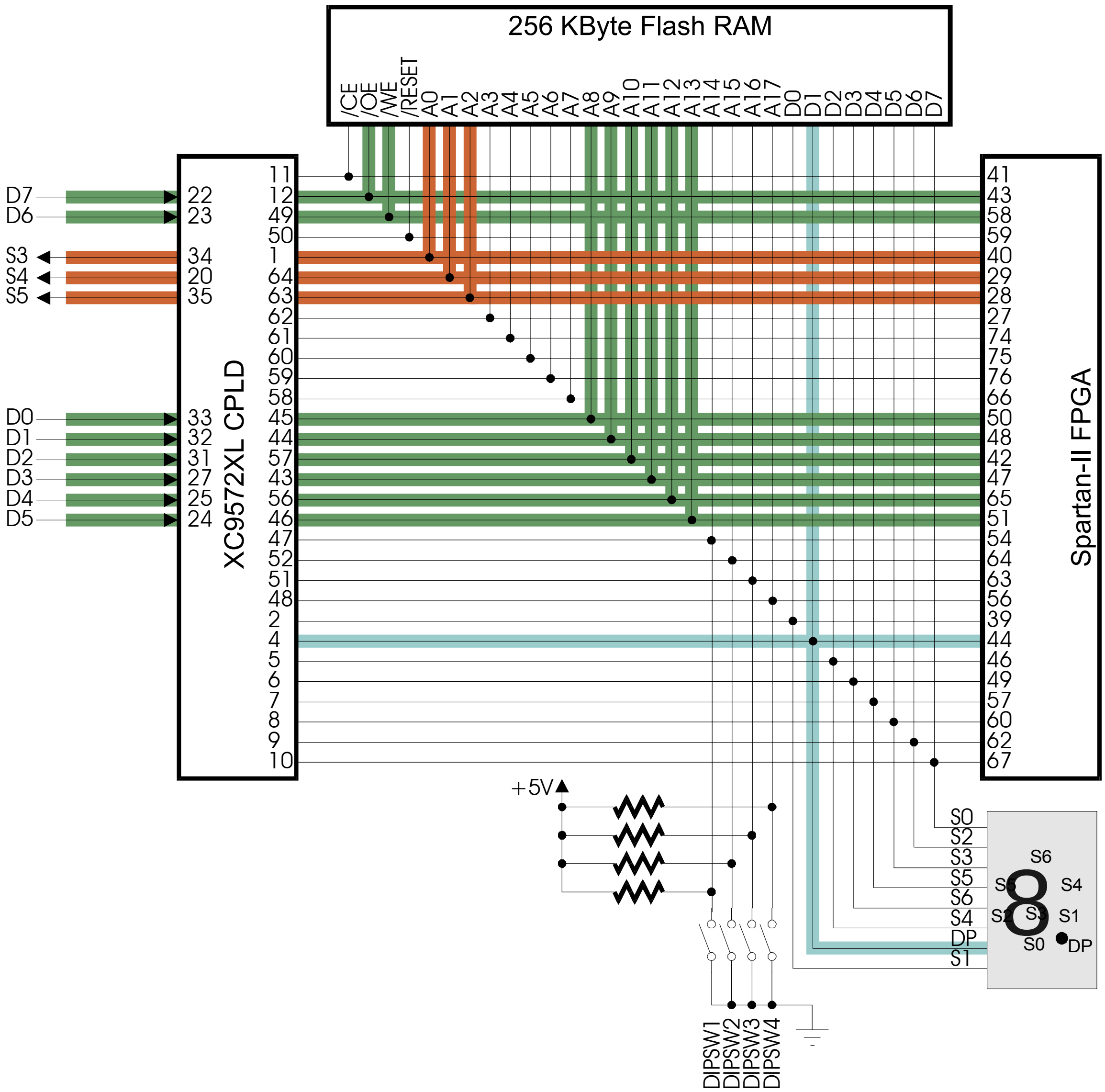
that control the loading of a bitstream. The CPLD uses the M0 input of the FPGA to select either the slave-serial or master-select configuration mode (M1 and M2 are already hard-wired to VCC and GND, respectively.) The CPLD can monitor the status of the bitstream download through the /INIT, DONE, and BSY/DOUT pins of the FPGA.

The CPLD also has access to the FPGA's JTAG pins: TCK, TMS, TDI, TDO. The TMS, TDI, and TDO pins share the connections with the BSY/DOUT, /CS, and /WR pins. With these connections, the CPLD can be programmed with an interface that allows configuration of the Spartan-II FPGA through the Xilinx iMPACT software. Jumper J9 allows the connection of status pin S7 to the general-purpose CPLD pin that also drives status pin S5. This is required by the iMPACT software so it can check for the presence of the downloading cable.



After the SpartanII FPGA is configured with a bitstream and the DONE pin goes high, the CPLD switches into a mode that connects the parallel port data and status pins to the FPGA. This lets you pass data to the FPGA over the parallel port data lines while

receiving data from the FPGA over the status lines. The connections between the FPGA and the parallel port are shown below.



The FPGA sends data back to the PC by driving logic levels onto pins 40, 29 and 28 which pass through the CPLD and onto the parallel port status lines S3, S4 and S5, respectively. Conversely, the PC sends data to the FPGA on parallel port data lines D0–D7 and the data passes through the CPLD and ends up on FPGA pins 50, 48, 42, 47, 65, 51, 58 and 43, respectively. The FPGA should never drive these pins unless it is accessing the Flash RAM otherwise the CPLD and/or the FPGA could be damaged. The CPLD can sense when the FPGA lowers the Flash RAM chip-enable and it will release the data lines so the FPGA can drive the address, output-enable and write-enable pins of the Flash RAM without contention.

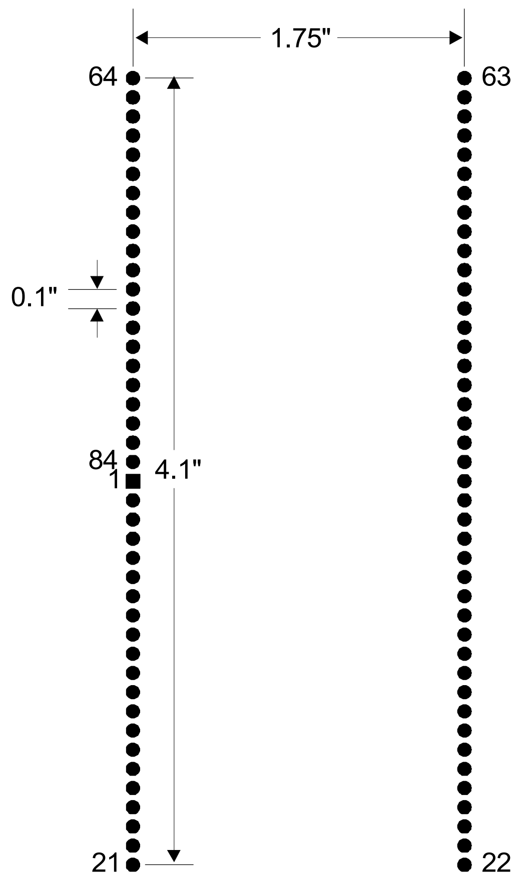
The CPLD also drives the decimal-point of the LED display to indicate when the FPGA is configured with a valid bitstream. Unless it is accessing the Flash RAM, the FPGA should never drive pin 44 to a low logic level or it may damage itself or the CPLD. But when the

FPGA lowers the Flash RAM chip-enable, the CPLD will stop driving the LED decimal-point to allow the FPGA access to data pin D1 of the Flash RAM.

For more details on how the CPLD manages the interface between the parallel port and the SpartanII FPGA both before and after device configuration, see the XSA Parallel Port Interface application note.

Prototyping Header

The pins of the FPGA are accessible through the 84-pin prototyping header on the underside of the XSA Board. Pin 1 of the header (denoted by a square pad) is located in the middle of the left-hand edge of the board and the remaining 83 pins are arranged counter-clockwise around the periphery. The physical dimensions of the prototyping header and the pin arrangement are shown below.



A subset of the 144 pins on the FPGA's TQFP package connects to the prototyping header. The number of the FPGA pin connected to a given header pin is printed next to the header pin on the board. This makes it easier to find a given FPGA pin when you want to connect it to an external system. While most of the FPGA pins are already used to support functions of the XSA Board, they can also be used to interface to external systems through the prototyping header. The FPGA pins can be grouped into the various categories shown below. (Pins denoted with * are useable as general-purpose I/O; pins denoted with ** can be used as general-purpose I/O only if the CPLD interface is

reprogrammed with the alternate parallel port interface stored in the dwnldpa2.svf file; pins with no marking cannot be used as general-purpose I/O at all.)

Configuration Pins (30*, 31*, 37, 38*, 39*, 44*, 46*, 49*, 57*, 60*, 62*, 67*, 68*, 69, 72, 106, 109, 111): These pins are used to load the SpartanII FPGA with a configuration bitstream. Some of these pins are dedicated to the configuration process and cannot be used as general-purpose I/O (37, 69, 72, 106, 109, 111). The rest can be used as general-purpose I/O after the FPGA is configured. If external logic is connected to these pins, you may have to disable it during the configuration process. The DONE pin (72) can be used for this purpose since it goes to a logic high only after the configuration process is completed.

Flash RAM Pins (27*, 28*, 29*, 39*, 40*, 41*, 42**, 43**, 44*, 46*, 47**, 48**, 49*, 50**, 51**, 54*, 56*, 57*, 58**, 59*, 60*, 62*, 63*, 64*, 65**, 66*, 67*, 74*, 75*, 76*): These pins are used by the FPGA to access the Flash RAM. They can be used for general-purpose I/O under the following conditions. When the FPGA is configured from the Flash, the CPLD drives all these pins so any external logic should be disabled using the DONE pin. Also, after the configuration, the Flash chip-enable (41) should be driven high to disable the Flash RAM so it doesn't drive the data bus pins. In addition, the standard parallel port interface loaded into the CPLD (dwnldpar.svf) will drive eight of the Flash RAM pins (42, 43, 47, 48, 50, 51, 58, 65) with the logic values found on the eight data lines of the parallel port. If this is not desired, then use the alternate parallel port interface (dwnldpa2.svf) which does not drive these pins.

VGA Pins (12*, 13*, 19*, 20*, 21*, 22*, 23*, 26*): When not used to drive a VGA monitor, these pins can be used for general-purpose I/O through the prototyping header. When used as I/O, the RED0–RED1 (12–13), GREEN0–GREEN1 (19–20) and BLUE0–BLUE1 (21–22) pairs have an impedance of approximately 1 KΩ between them due to the presence of the resistor-ladder DAC circuitry.

PS/2 Pins (93*, 94*): When not used to access the PS/2 keyboard/mouse port, these pins can be used as general-purpose I/O through the prototyping header.

Global Clock Pins (15*, 18*): These pins can be used as global clock inputs or general-purpose inputs. They cannot be used as outputs.

Free Pins (77*, 78*, 79*, 80*, 83*, 84*, 85*, 86*, 87*): These pins are not connected to any other devices on the XSA Board so they can be used without restrictions as general-purpose I/O through the prototyping header.

JTAG Pins (2, 32, 34, 142): These pins are used to access the JTAG features of the FPGA. They cannot be used as general-purpose I/O pins.

A

XSA Pin Connections

The following tables list the pin numbers of the Spartan-II FPGA and the XC9572XL CPLD along with the pins of the other chips that they connect to on the XSA Board. The columns of the table are arranged as follows:

Column 1 lists the Spartan-II FPGA pin. It is left blank if there is no connection to the FPGA for this function. Pins marked with * are useable as general-purpose I/O through the prototyping header; pins denoted with ** can be used as general-purpose I/O only if the CPLD interface is reprogrammed with the alternate parallel port interface stored in the dwnldpa2.svf file; pins with no marking cannot be used as general-purpose I/O at all.

Column 2 lists the XC9572XL CPLD pin. It is left blank if there is no connection to the CPLD for this function.

Column 3 lists the pins of other devices on the XSA Board that are connected to the associated FPGA and/or CPLD pin.

Column 4 lists the pin of the XSA prototyping header that is connected to the associated FPGA and/or CPLD pin.

Columns 5–7 list the pins of devices on the Xstend Board that will connect to the FPGA and/or CPLD when the XSA Board is inserted into an Xstend Board. The first table lists the connections for the XST-2 Board while the second table lists the connections for the older XST-1.3 Board.

FPGA	CPLD	XSA Function	Proto. Pin	XST-2.x Functions		
1		+3.3V	54	+3.3V		
2	13	SPARTAN-TCK	16			
3		SDRAM-A7				
4		SDRAM-A1				
5		SDRAM-A6				
6		SDRAM-A2				
7		SDRAM-A5				
8		GND	52	GND		
9		+2.5V	22			
10		SDRAM-A3				
11		SDRAM-A4				
12*		VGA-RED0	27			
13*		VGA-RED1	28			
15*		SPARTAN-GCK3	31			
18*		SPARTAN-GCK2	1			
19*		VGA-GREEN0	29			
20*		VGA-GREEN1	32			
21*		VGA-BLUE0	33			
22*		VGA-BLUE1	34			
23*		VGA-/HSYNC	36		PUSHB4	
26*		VGA-/VSYNC	37		PUSHB3	
27*	62	FLASH-A3	50	RAM-A0	LED2-B	IDE_DMARQ
28*	63	FLASH-A2, *PARPORT-S5	51	RAM-A10	LED2-E	/USB_INT
29*	64	FLASH-A1, *PARPORT-S4	56	RAM-A11	LED2-G	USB_SUSPEND
30*	19	SPARTAN-/WRITE	69		DIPSW1	
31*	15	SPARTAN-CS	68			/IDE_RESET
32	15*	SPARTAN-TDI	15			
34	19*	SPARTAN-TDO	30			
37	16	SPARTAN-CCLK	73			
38*	18	SPARTAN-DOUT/BSY	45	RAM-A1	LED2-DP	/IDE_DMACK
39*	2	FLASH-D0,DIN/D0,LED-S1	71	RAM-A16	BARLED-9	IDE_IORDY
40*	1	FLASH-A0, *PARPORT-S3	57	RAM-A9	LED2-C	IDE_INTRQ
41*	11	FLASH-/CE	65			
42**	57	FLASH-A10, *PARPORT-D2	58	RAM-A8	LED2-F	IDE_D8
43**	12	FLASH-/OE, *PARPORT-D7	61	/RAM-OE		IDE_D9
44*	4	FLASH-D1,LED-DP	40	RAM-D6	BARLED-2	IDE_D1
46*	5	FLASH-D2,LED-S4	39	RAM-D5	BARLED-3	IDE_D2
47**	43	FLASH-A11, *PARPORT-D3	59	RAM-A13	LED2-D	IDE_D10
48**	44	FLASH-A9, *PARPORT-D1	60	RAM-A15	LED2-A	IDE_D11
49*	6	FLASH-D3,LED-S6	38	RAM-D4	BARLED-4	IDE_D3
50**	45	FLASH-A8, *PARPORT-D0	78	RAM-A14	LED1-G	IDE_D12
51**	46	FLASH-A13, *PARPORT-D5	79	RAM-A12	LED1-B	IDE_D13
54*	47	FLASH-A14,DIPSW1A	82	RAM-A7	LED1-F	/IDE_CS0
56*	48	FLASH-A17,DIPSW1D	83	RAM-A6	LED1-A	/IDE_CS1
57*	7	FLASH-D4,LED-S5	35	RAM-D3	BARLED-5	IDE_D4
58**	49	FLASH-/WE, *PARPORT-D6	62	/RAM-WE	DIPSW2	IDE_D14
59*	50	FLASH-/RESET	66	AUDIO_LRCK	BARLED-10	
60*	8	FLASH-D5,LED-S3	80	RAM-D0	BARLED-7	IDE_D6,RS232_RD
62*	9	FLASH-D6,LED-S2	81	RAM-D1	BARLED-6	IDE_D5, RS232_CTS
63*	51	FLASH-A16,DIPSW1C	84	RAM-A5	LED1-DP	IDE_DA2
64*	52	FLASH-A15,DIPSW1B	3	RAM-A4	LED1-D	IDE_DA0
65**	56	FLASH-A12, *PARPORT-D4	4	RAM-A3	LED1-C	IDE_D15
66*	58	FLASH-A7	5	RAM-A2	DIPSW5	IDE_DA1
67*	10	FLASH-D7,LED-S0	10	RAM-D2	BARLED-8	IDE_D7
68*	38	SPARTAN-/INIT	41	RAM-D7	BARLED-1	IDE_D0
69	39	SPARTAN-/PROGRAM	55		PUSHB1	
72	40	SPARTAN-DONE	53			
74*	61	FLASH-A4	70	AUDIO_SDTI	DIPSW3	
75*	60	FLASH-A5	77	AUDIO_SCLK	DIPSW4	
76*	59	FLASH-A6	6	AUDIO_SDTO	LED1-E	
77*			9	AUDIO_MCLK	DIPSW6	
78*		PARPORT-S6	67		PUSHB2	
79*			7	/RAM_CE	DIPSW8	
80*			8		DIPSW7	RS232_RTS

FPGA	CPLD	XSA Function	Proto. Pin	XST-2.x Functions		
83*			18			RS232_TD
84*			19			USB_SCL
85*			20			USB_SDA
86*			23			/IDE_DIOR
87*			24			/IDE_DIOW
88	42	MASTER_CLK	13	MASTER_CLK		
91		SDRAM-CLK				
93*		PS2-DATA,PUSHBUTTON	25			
94*		PS2-CLK	26			
95		SDRAM-Q0				
96		SDRAM-Q15				
99		SDRAM-Q1				
100		SDRAM-Q14				
101		SDRAM-Q2				
102		SDRAM-Q13				
103		SDRAM-Q3				
106		SPARTAN-M2	12			
109	36	SPARTAN-M0	14			
111		SPARTAN-M1	21			
112		SDRAM-Q12				
113		SDRAM-Q4				
114		SDRAM-Q11				
115		SDRAM-Q5				
116		SDRAM-Q10				
117		SDRAM-Q6				
118		SDRAM-Q9				
120		SDRAM-Q7				
121		SDRAM-Q8				
122		SDRAM-QML				
123		SDRAM-/WE				
124		SDRAM-QMH				
126		SDRAM-/CAS				
129		SDRAM-CLK				
130		SDRAM-/RAS				
131		SDRAM-CKE				
132		SDRAM-/CS				
133		SDRAM-A12				
134		SDRAM-BA0				
136		SDRAM-A11				
137		SDRAM-BA1				
138		SDRAM-A9				
139		SDRAM-A10				
140		SDRAM-A8				
141		SDRAM-A0				
142	18*	SPARTAN-TMS	17			
	30	PARPORT-C1,CPLD-TCK				
	29	PARPORT-C2,CPLD-TMS				
	28	PARPORT-C3,CPLD-TDI				
	33	PARPORT-D0				
	32	PARPORT-D1				
	31	PARPORT-D2				
	27	PARPORT-D3				
	25	PARPORT-D4				
	24	PARPORT-D5				
	23	PARPORT-D6				
	22	PARPORT-D7				
	34	PARPORT-S3				
	20	PARPORT-S4				
	35	PARPORT-S5				
	53	PARPORT-S7,CPLD-TDO				
	17	PROG-OSC				
			64	Osc-In		USB_CLKOUT

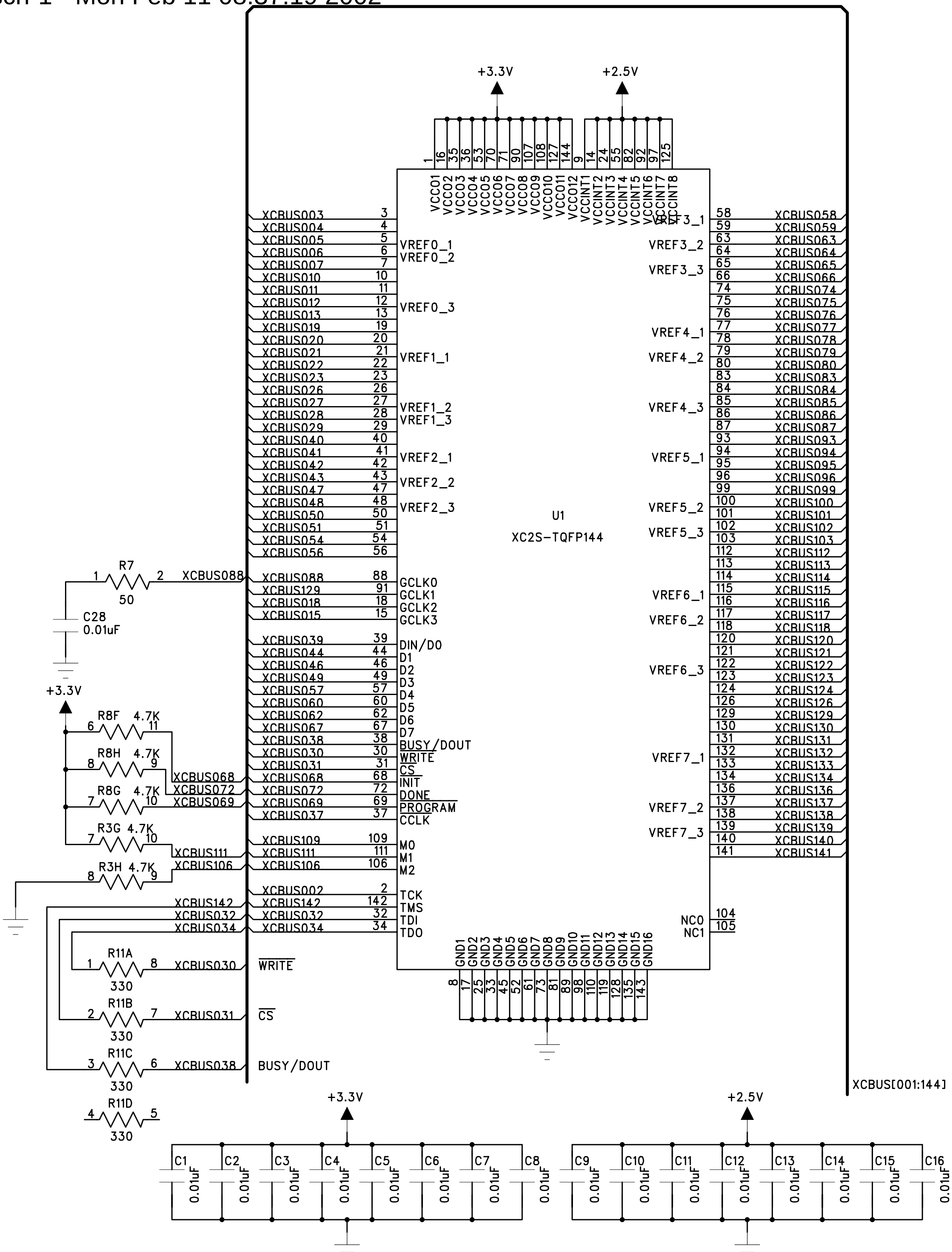
FPGA	CPLD	XSA Function	Proto. Pin	XST-1.3.x Functions		
1		+3.3V	54	+3.3V		
2	13	SPARTAN-TCK	16			Xchecker-TCK
3		SDRAM-A7				
4		SDRAM-A1				
5		SDRAM-A6				
6		SDRAM-A2				
7		SDRAM-A5				
8		GND	52	GND		
9		+2.5V	22			
10		SDRAM-A3				
11		SDRAM-A4				
12*		VGA-RED0	27			
13*		VGA-RED1	28	RAM-A15	RLED-/DP	
15*		SPARTAN-GCK3	31			
18*		SPARTAN-GCK2	1			
19*		VGA-GREEN0	29			
20*		VGA-GREEN1	32			Xchecker-RT
21*		VGA-BLUE0	33			
22*		VGA-BLUE1	34			
23*		VGA-/HSYNC	36			
26*		VGA-/VSYNC	37		Pushbutton-/RESET	
27*	62	FLASH-A3	50	RAM-A12	RLED-/S4	
28*	63	FLASH-A2, *PARPORT-S5	51	RAM-A10	RLED-/S2	
29*	64	FLASH-A1, *PARPORT-S4	56	RAM-A11	RLED-/S3	
30*	19	SPARTAN-/WRITE	69	PS2/DATA	DIPSW8	
31*	15	SPARTAN-CS	68	PS2/CLK		
32	15*	SPARTAN-TDI	15			Xchecker-TDI
34	19*	SPARTAN-TDO	30			Xchecker-RD
37	16	SPARTAN-CCLK	73			Xchecker-CCLK
38*	18	SPARTAN-DOUT/BSY	45			
39*	2	FLASH-D0,DIN/D0,LED-S1	71			Xchecker-DIN
40*	1	FLASH-A0, *PARPORT-S3	57	RAM-A9	RLED-/S1	
41*	11	FLASH-/CE	65	RAM-/CE		
42**	57	FLASH-A10, *PARPORT-D2	58	RAM-A13	RLED-/S5	
43**	12	FLASH-/OE, *PARPORT-D7	61	RAM-/OE		
44*	4	FLASH-D1,LED-DP	40	RAM-D1	BARLED-2	
46*	5	FLASH-D2,LED-S4	39	RAM-D2	BARLED-3	
47**	43	FLASH-A11, *PARPORT-D3	59	RAM-A8	RLED-/S0	
48**	44	FLASH-A9, *PARPORT-D1	60	RAM-A14	RLED-/S6	
49*	6	FLASH-D3,LED-S6	38	RAM-D3	BARLED-4	
50**	45	FLASH-A8, *PARPORT-D0	78	RAM-A3	LLED-/S3	
51**	46	FLASH-A13, *PARPORT-D5	79	RAM-A4	LLED-/S4	
54*	47	FLASH-A14,DIPSW1A	82	RAM-A5	LLED-/S5	
56*	48	FLASH-A17,DIPSW1D	83	RAM-A6	LLED-/S6	
57*	7	FLASH-D4,LED-S5	35	RAM-D4	BARLED-5	
58**	49	FLASH-/WE, *PARPORT-D6	62	RAM-WE		
59*	50	FLASH-/RESET	66	CODEC-LRCK	DIPSW7	
60*	8	FLASH-D5,LED-S3	80	RAM-D6	BARLED-7	
62*	9	FLASH-D6,LED-S2	81	RAM-D5	BARLED-6	
63*	51	FLASH-A16,DIPSW1C	84	RAM-A7	LLED-/DP	
64*	52	FLASH-A15,DIPSW1B	3	RAM-A0	LLED-/S0	
65**	56	FLASH-A12, *PARPORT-D4	4	RAM-A1	LLED-/S1	
66*	58	FLASH-A7	5	RAM-A2	LLED-/S2	
67*	10	FLASH-D7,LED-S0	10	RAM-D7	BARLED-8	
68*	38	SPARTAN-/INIT	41	RAM-D0	BARLED-1	Xchecker-INIT
69	39	SPARTAN-/PROGRAM	55		Pushbutton-/PROGRAM	Xchecker-PROG
72	40	SPARTAN-DONE	53			Xchecker-DONE
74*	61	FLASH-A4	70	CODEC-SDIN	DIPSW6	
75*	60	FLASH-A5	77	CODEC-SCLK	DIPSW5	
76*	59	FLASH-A6	6	CODEC-SDOUT	DIPSW4	
77*			9	CODEC-MCLK	DIPSW3	Xchecker-CLKO
78*		PARPORT-S6	67	VGA-/VSYNC	Pushbutton-/SPARE	
79*			7	RAM-/LCE	DIPSW1	Xchecker-TRIG
80*			8	RAM-/RCE	DIPSW2	Xchecker-RST

FPGA	CPLD	XSA Function	Proto. Pin	XST-1.3.x Functions		
83*			18	VGA-RED1		
84*			19	VGA-/HSYNC		
85*			20	VGA-GREEN1		
86*			23	VGA-RED0		
87*			24	VGA-GREEN0		
88	42	MASTER_CLK	13	MASTER_CLK		Xchecker-CLKI
91		SDRAM-CLK				
93*		PS2-DATA,PUSHBUTTON	25	VGA-BLUE0		
94*		PS2-CLK	26	VGA-BLUE1		
95		SDRAM-Q0				
96		SDRAM-Q15				
99		SDRAM-Q1				
100		SDRAM-Q14				
101		SDRAM-Q2				
102		SDRAM-Q13				
103		SDRAM-Q3				
106		SPARTAN-M2	12			
109	36	SPARTAN-M0	14			
111		SPARTAN-M1	21			
112		SDRAM-Q12				
113		SDRAM-Q4				
114		SDRAM-Q11				
115		SDRAM-Q5				
116		SDRAM-Q10				
117		SDRAM-Q6				
118		SDRAM-Q9				
120		SDRAM-Q7				
121		SDRAM-Q8				
122		SDRAM-QML				
123		SDRAM-/WE				
124		SDRAM-QMH				
126		SDRAM-/CAS				
129		SDRAM-CLK				
130		SDRAM-/RAS				
131		SDRAM-CKE				
132		SDRAM-/CS				
133		SDRAM-A12				
134		SDRAM-BA0				
136		SDRAM-A11				
137		SDRAM-BA1				
138		SDRAM-A9				
139		SDRAM-A10				
140		SDRAM-A8				
141		SDRAM-A0				
142	18*	SPARTAN-TMS	17			Xchecker-TMS
	30	PARPORT-C1,CPLD-TCK				
	29	PARPORT-C2,CPLD-TMS				
	28	PARPORT-C3,CPLD-TDI				
	33	PARPORT-D0				
	32	PARPORT-D1				
	31	PARPORT-D2				
	27	PARPORT-D3				
	25	PARPORT-D4				
	24	PARPORT-D5				
	23	PARPORT-D6				
	22	PARPORT-D7				
	34	PARPORT-S3				
	20	PARPORT-S4				
	35	PARPORT-S5				
	53	PARPORT-S7,CPLD-TDO				
	17	PROG-OSC				
			64	Osc-In		

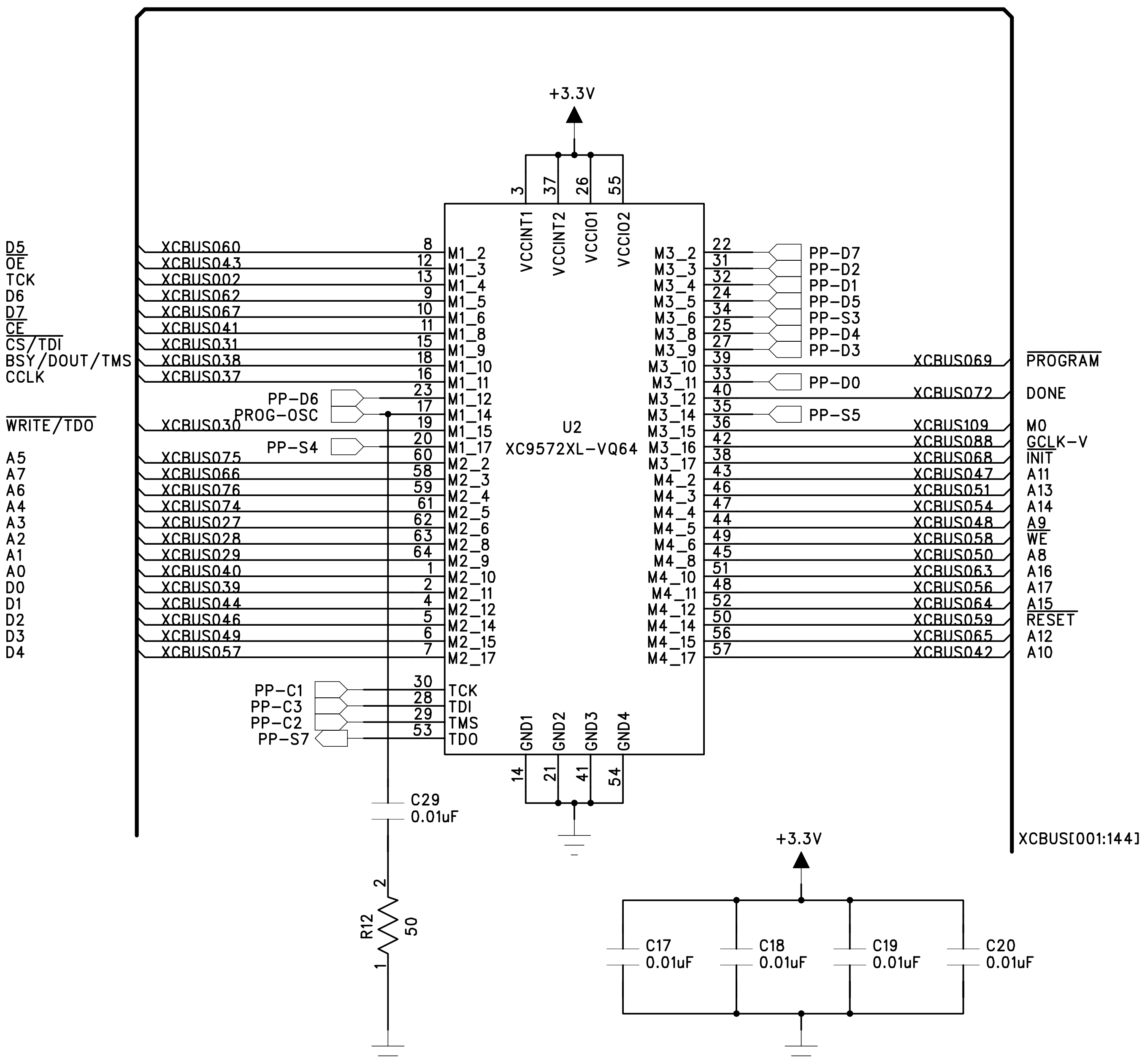
B

XSA Schematics

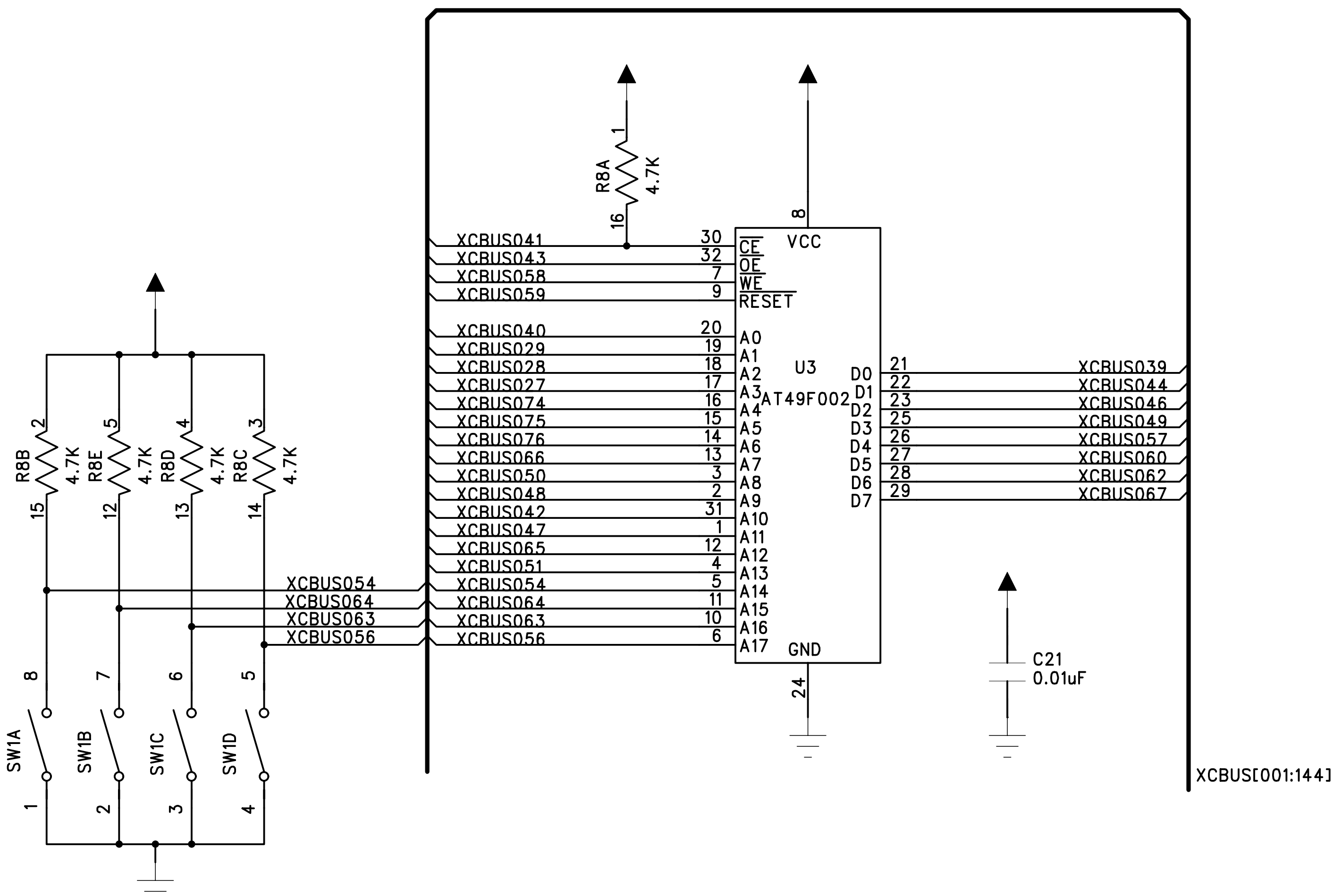
The following pages show the detailed schematics for the XSA Board.



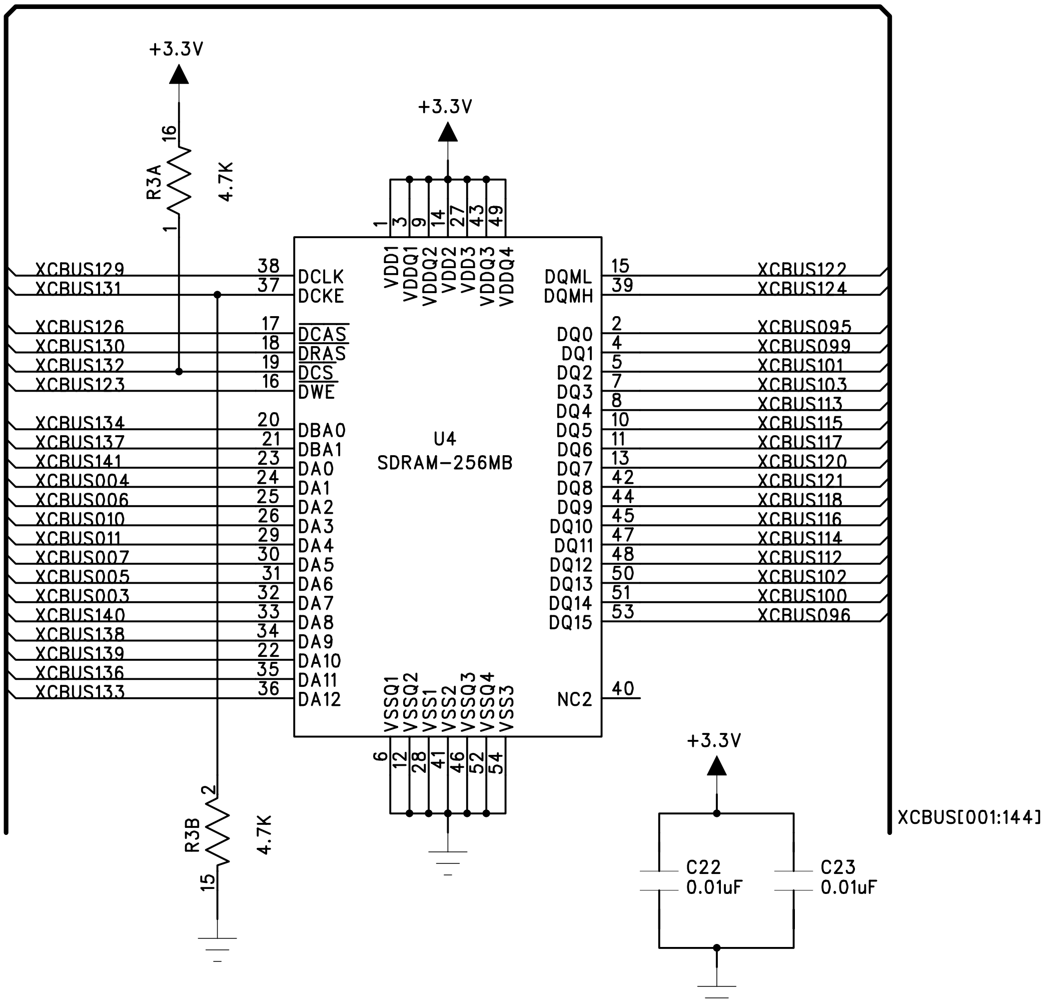
COMPANY: XESS Corporation	
TITLE: XSA Board Spartan FPGA	
DRAWN:	DATED:
REV: V1.2	
RELEASED:	DATED:
SHEET:	OF



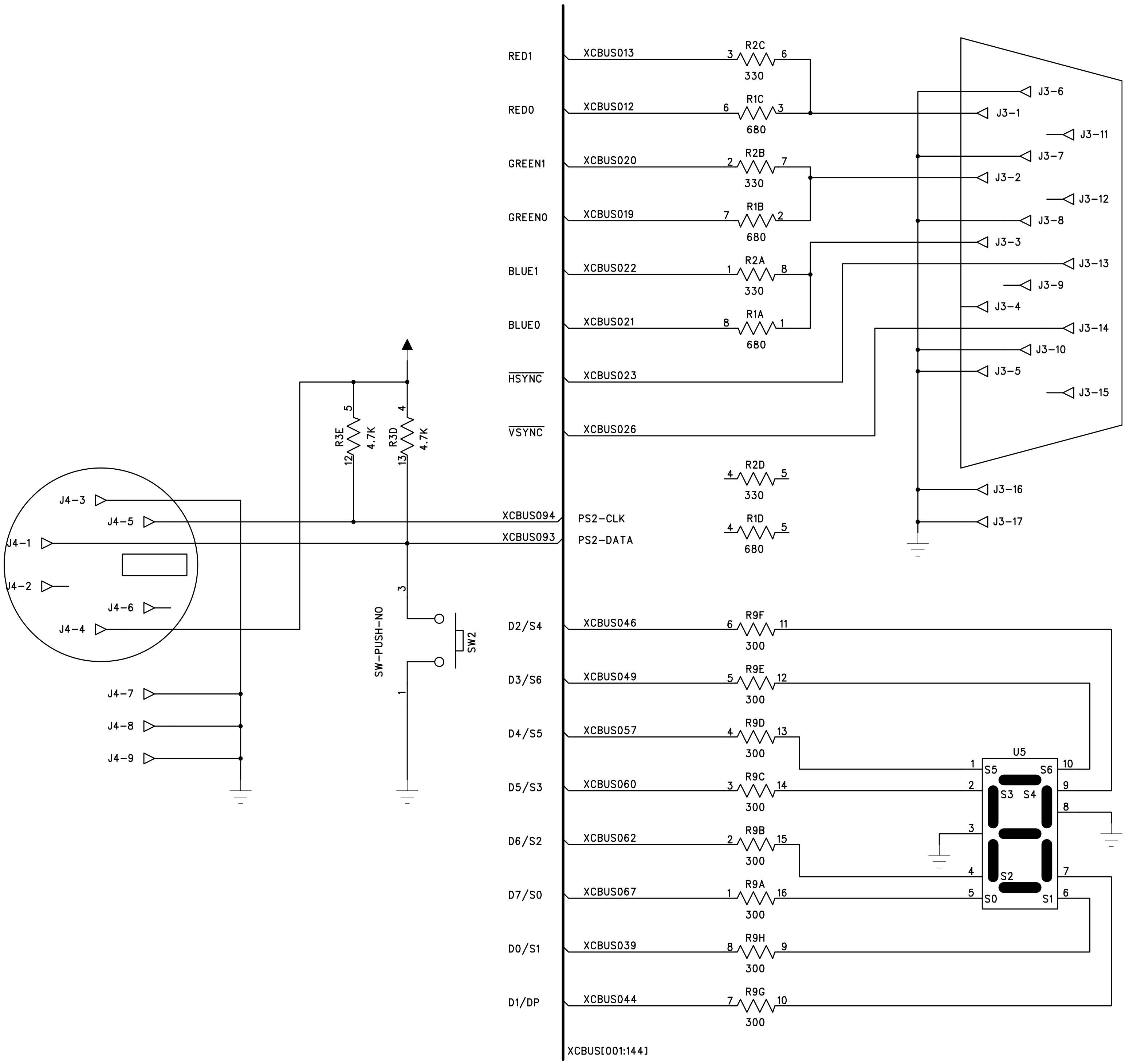
COMPANY: XESS Corporation		
TITLE: XSA Board CPLD Interface		
DRAWN:	DATED:	REV: V1.2
RELEASED:	DATED:	SHEET: OF



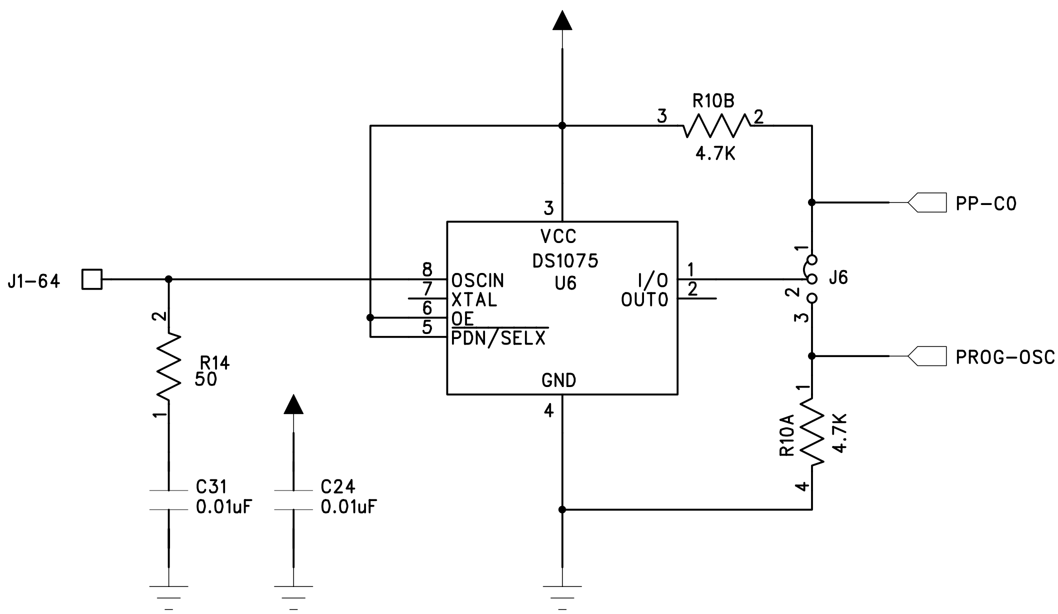
COMPANY: XESS Corporation			
TITLE: XSA Board Flash RAM			
DRAWN:	DATED:	REV: V1.2	
RELEASED:	DATED:	SHEET:	OF



COMPANY:	XESS Corporation		
TITLE:	XSA Board Sync. DRAM		
DRAWN:	DATED:	REV:	V1.2
RELEASED:	DATED:	SHEET:	OF

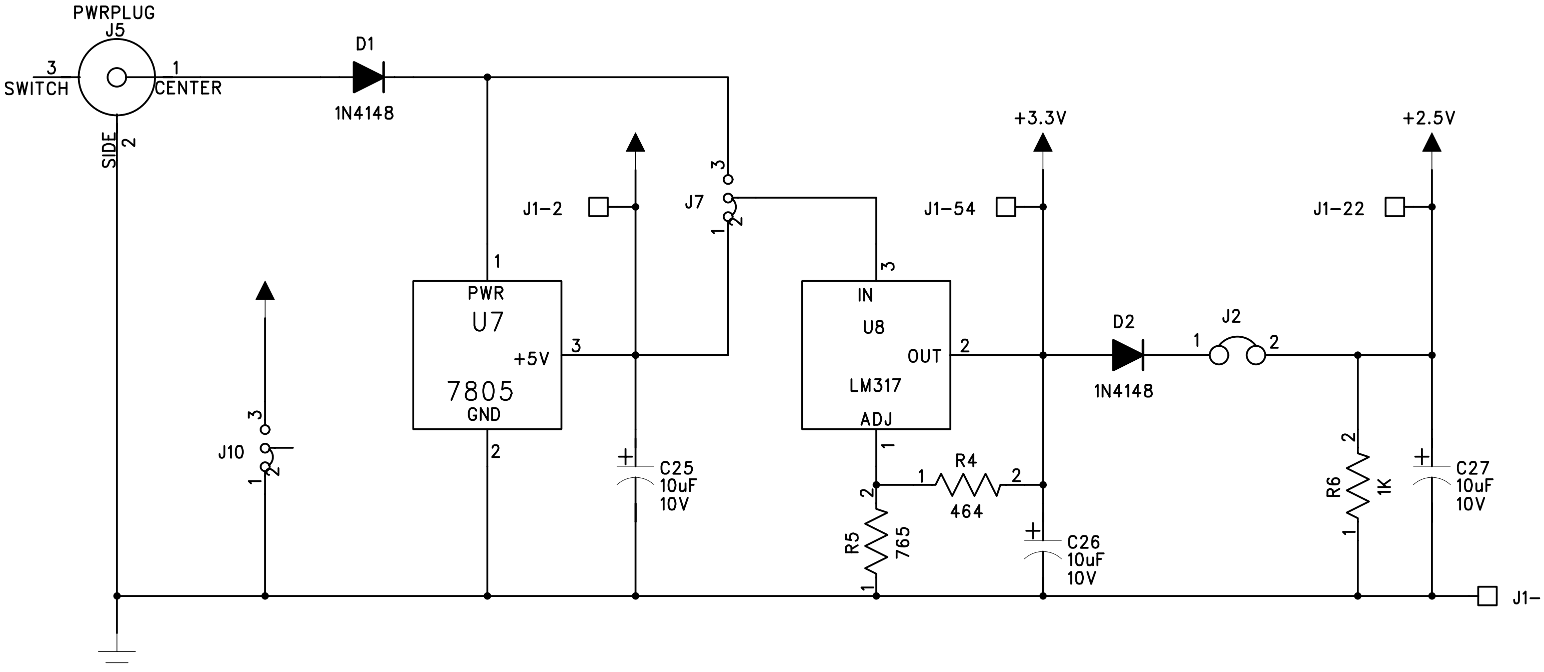


COMPANY: XESS Corporation		
TITLE: XSA Board PS/2 Port, VGA Port, LED		
DRAWN:	DATED:	REV: V1.2
RELEASED:	DATED:	SHEET: OF

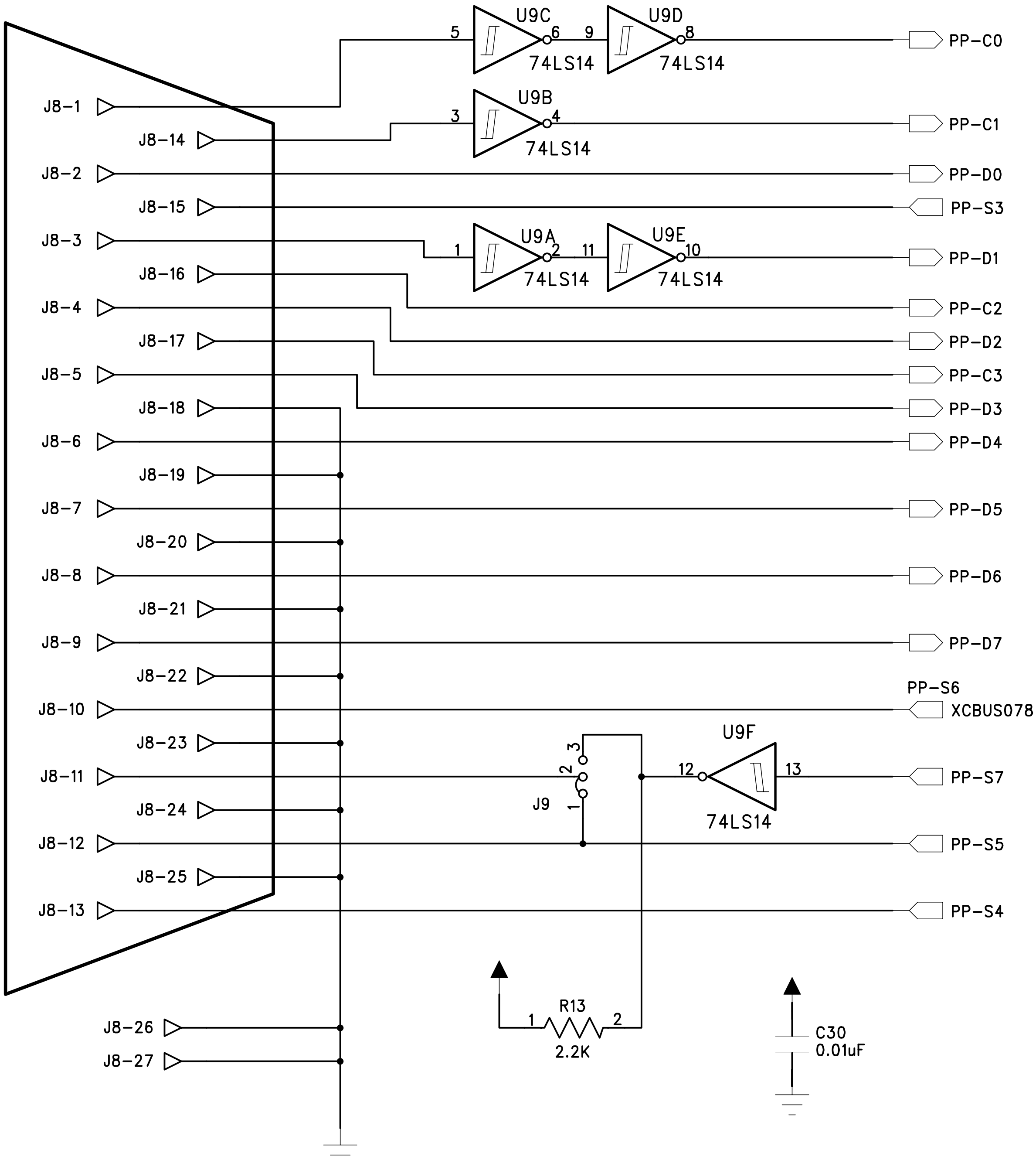


COMPANY: XESS Corporation		
TITLE: XSA Board Programmable Oscillator		
DRAWN:	DATED:	REV: V1.2
RELEASED:	DATED:	SHEET: OF

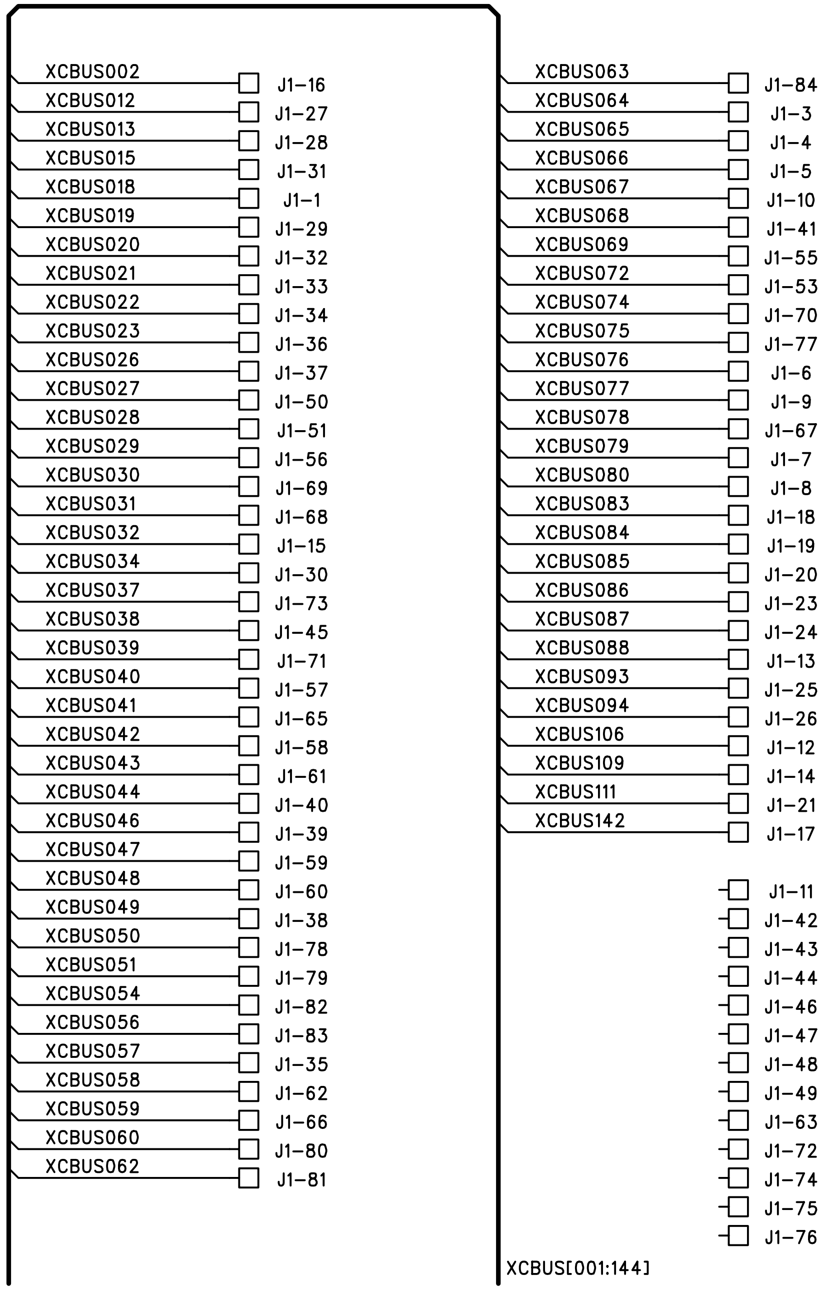
xsa1_2.sch-7 - Mon Feb 11 08:37:20 2002



COMPANY:	XESS Corporation
TITLE:	XSA Board Regulated Power Supplies
DRAWN:	DATED:
REV:	V1.2
RELEASED:	DATED:
SHEET:	OF



COMPANY: XESS Corporation			
TITLE: XSA Board Parallel Port Interface			
DRAWN:	DATED:	REV: V1.2	
RELEASED:	DATED:	SHEET:	OF



COMPANY:	XESS Corporation
TITLE:	XSA Board Prototyping Header
DRAWN:	DATED:
REV:	V1.2
RELEASED:	DATED:
SHEET:	OF